NEPAL ENGINEERING COUNCIL (NEC)

Computer Engineering Licensure Examination — Academic & Examination Support Series

COMPREHENSIVE SOLUTIONS, DETAILED REASONING & ANALYSIS GUIDE (MODEL SET 01)

MASTER ANSWER KEY MATRIX (QUESTIONS 01 – 100)

Q01	Q02	Q03	Q04	Q05	Q06	Q07	Q08	Q09	Q10
C	B	C	B	B	C	B	B	B	A
Q11	Q12	Q13	Q14	Q15	Q16	Q17	Q18	Q19	Q20
A	C	B	C	B	B	B	C	C	D
Q21	Q22	Q23	Q24	Q25	Q26	Q27	Q28	Q29	Q30
A	B	B	B	B	B	C	B	B	B
Q31	Q32	Q33	Q34	Q35	Q36	Q37	Q38	Q39	Q40
C	B	A	B	B	B	B	B	B	B
Q41	Q42	Q43	Q44	Q45	Q46	Q47	Q48	Q49	Q50
B	B	D	A	B	B	B	B	A	B
Q51	Q52	Q53	Q54	Q55	Q56	Q57	Q58	Q59	Q60
B	B	C	B	A	C	B	B	A	C
Q61	Q62	Q63	Q64	Q65	Q66	Q67	Q68	Q69	Q70
B	A	B	C	B	B	B	A	C	C
Q71	Q72	Q73	Q74	Q75	Q76	Q77	Q78	Q79	Q80
B	C	D	B	B	C	C	B	B	B
Q81	Q82	Q83	Q84	Q85	Q86	Q87	Q88	Q89	Q90
C	A	B	A	B	C	B	B	D	A
Q91	Q92	Q93	Q94	Q95	Q96	Q97	Q98	Q99	Q100
B	B	B	A	B	A	B	B	B	A

SECTION I: CONCEPT OF BASIC ELECTRICAL AND ELECTRONICS ENGINEERING

Solutions Q.01 – Q.10

Question 01

1.1 Basic Concept & Star-Delta Conversion

Three identical resistors, each having a resistance of 12 Ω, are connected in a Delta (Δ) network. What is the equivalent resistance of each branch when transformed into an equivalent Star (Y) network?

✓ Correct Answer: **(C) 4 Ω**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In Star-Delta transformation, the resistance of each branch in a Star network (R_Y) formed from three identical Delta resistors (R_Δ) is given by:

$$R_Y = \frac{R_\Delta \times R_\Delta}{R_\Delta + R_\Delta + R_\Delta} = \frac{R_\Delta^2}{3R_\Delta} = \frac{R_\Delta}{3}$$

Substituting $R_\Delta = 12\,\Omega$:

$$R_Y = \frac{12\,\Omega}{3} = 4\,\Omega$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 36 Ω corresponds to converting from Star to Delta ($3 \times 12\,\Omega = 36\,\Omega$), which is the inverse operation.

Option (B): Incorrect. 12 Ω assumes no change in resistance, which violates electrical equivalence.

Option (C): Correct. Each Star branch resistance equals one-third of the balanced Delta branch resistance ($12/3 = 4\,\Omega$).

Option (D): Incorrect. 6 Ω represents two 12 Ω resistors in parallel, not a balanced three-phase Star transformation.

HIGH-YIELD EXAM TAKEAWAY: For balanced systems: $R_Y = \frac{R_\Delta}{3}$ and $R_\Delta = 3R_Y$.

Question 02 1.2 Network Theorems: Thevenin's Theorem

When determining the Thevenin equivalent resistance (R_{th}) looking into the load terminals of a linear bilateral network, how must the independent sources inside the network be deactivated?

✓ **Correct Answer: (B) Independent voltage sources are short-circuited; independent current sources are open-circuited**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

To determine Thevenin's equivalent resistance (R_{th}), all independent sources must be turned off (set to zero):
1. An ideal voltage source has zero internal impedance; setting its voltage to zero ($V = 0$) is equivalent to a **short circuit**.
2. An ideal current source has infinite internal impedance; setting its current to zero ($I = 0$) is equivalent to an **open circuit**.
Dependent sources, if present, must remain active and require test voltage/current injection (V_t / I_t).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. This reverses the physical nature of ideal voltage and current sources.
Option (B): Correct. Voltage sources are replaced by short circuits ($0\ \text{V}$), and current sources are replaced by open circuits ($0\ \text{A}$).
Option (C): Incorrect. Short-circuiting a current source forces a non-zero current path rather than cutting it to zero.
Option (D): Incorrect. Open-circuiting a voltage source creates infinite impedance rather than zero impedance.

HIGH-YIELD EXAM TAKEAWAY: Independent source deactivation rule: Voltage Source $\rightarrow$ Short Circuit ($0\ \text{V}$); Current Source $\rightarrow$ Open Circuit ($0\ \text{A}$).

Question 03 1.2 Maximum Power Transfer Theorem

A DC source has an open-circuit voltage of $V_{th} = 24\ \text{V}$ and an internal Thevenin resistance of $R_{th} = 6\,\Omega$. What is the maximum power that can be delivered to a variable load resistor R_L, and what is the electrical transmission efficiency at this operating point?

✓ Correct Answer: **(C) 24 W at 50% efficiency**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

According to the Maximum Power Transfer Theorem, maximum power is transferred to the load when the load resistance equals the Thevenin source resistance ($R_L = R_{th} = 6\,\Omega$).
The maximum power delivered to the load is:

$$P_{max} = \frac{V_{th}^2}{4R_{th}} = \frac{24^2}{4 \times 6} = \frac{576}{24} = 24\ \text{W}$$

The total power generated by the source is:

$$P_{total} = I^2(R_{th} + R_L) = \left(\frac{24}{6+6}\right)^2 \times 12 = (2\ \text{A})^2 \times 12 = 48\ \text{W}$$

Efficiency:

$$\eta = \frac{P_{load}}{P_{total}} \times 100\% = \frac{24\ \text{W}}{48\ \text{W}} \times 100\% = 50\%$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 96 W uses V^2/R without accounting for the voltage divider, and efficiency can never be 100% when source resistance is non-zero.
Option (B): Incorrect. 48 W is the total power generated by the source, not the power delivered to the load.
Option (C): Correct. $P_{max} = 24\ \text{W}$ and efficiency $\eta = 50\%$.
Option (D): Incorrect. 12 W results from an arithmetic error in the denominator.

HIGH-YIELD EXAM TAKEAWAY: At maximum power transfer: $R_L = R_{th}$, $P_{max} = \frac{V_{th}^2}{4R_{th}}$, and efficiency is strictly 50%.

Question 04 1.2 R-L-C Circuits & Resonance

In a series RLC AC circuit operating precisely at its resonant frequency (f_0), which of the following statements is strictly TRUE?

✓ **Correct Answer: (B) Total circuit impedance is at its minimum and purely resistive ($Z = R$), resulting in a unity power factor**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In a series RLC circuit, total impedance is:

$$Z = \sqrt{R^2 + (X_L - X_C)^2}$$

At resonance ($f_0 = \frac{1}{2\pi\sqrt{LC}}$), inductive reactance equals capacitive reactance ($X_L = X_C$). Therefore, the reactive component cancels completely:

$$Z_{min} = R$$

Because the circuit is purely resistive at resonance, the phase angle between voltage and current is zero ($\phi = 0°$), yielding a unity power factor ($\cos\phi = 1.0$) and maximum current ($I_{max} = V/R$).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $X_L > X_C$ occurs above resonance, not at resonance.

Option (B): Correct. $X_L = X_C$, so impedance is minimized ($Z = R$), current is maximized, and power factor is unity.

Option (C): Incorrect. Maximum impedance and minimum current describe parallel resonance (anti-resonance), not series resonance.

Option (D): Incorrect. Pure capacitive behavior requires $R = 0$ and operating well below resonance.

HIGH-YIELD EXAM TAKEAWAY: Series resonance $\implies X_L = X_C$, $Z_{min} = R$, $I_{max} = \frac{V}{R}$, and $\cos\phi = 1$.

Question 05 1.3 Alternating Current Fundamentals: 3-Phase Systems

In a balanced three-phase Star (Wye) connected AC electrical system, what is the exact mathematical relationship between the line parameters (V_L, I_L) and phase parameters (V_{ph}, I_{ph})?

✓ **Correct Answer: (B) $V_L = \sqrt{3}V_{ph}$ and $I_L = I_{ph}$**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In a Star (Wye) connection:

1. The line conductors connect directly to each phase winding in series, so line current equals phase current: $I_L = I_{ph}$.
2. The line-to-line voltage is the phasor difference between two phase voltages separated by $120°$:

$$V_L = 2V_{ph}\cos(30°) = \sqrt{3}V_{ph}\angle 30°$$

In contrast, in a Delta (Δ) connection, $V_L = V_{ph}$ and $I_L = \sqrt{3}I_{ph}$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. This describes a balanced Delta (Δ) connected system.

Option (B): Correct. In Star connection: $V_L = \sqrt{3}V_{ph}$ and $I_L = I_{ph}$.

Option (C): Incorrect. $\sqrt{3}$ factor applies to voltage in Star and to current in Delta, never both simultaneously.

Option (D): Incorrect. Line voltage is larger than phase voltage by $\sqrt{3}$, not smaller.

HIGH-YIELD EXAM TAKEAWAY: Star connection: $V_L = \sqrt{3}V_{ph}$, $I_L = I_{ph}$. Delta connection: $V_L = V_{ph}$, $I_L = \sqrt{3}I_{ph}$.

Question 06 1.4 Semiconductor Devices: Diodes

How does the reverse saturation current (I_0) of a standard silicon P-N junction diode change with an increase in junction operating temperature?

✓ Correct Answer: **(C) It approximately doubles for every 10°C rise in temperature**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Reverse saturation current (I_0) in a P-N junction diode is generated by thermally generated minority charge carriers. The empirical relationship governing I_0 as a function of temperature is:

$$I_0(T_2) = I_0(T_1) \times 2^{\frac{T_2 - T_1}{10}}$$

For both silicon and germanium diodes, I_0 approximately doubles for every $10^{\circ}\mathrm{C}$ rise in operating junction temperature, which is why reverse leakage current becomes significant at elevated temperatures.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Reverse leakage current increases exponentially, not decreases.
Option (B): Incorrect. Doping sets majority carrier density; minority carrier generation depends heavily on thermal energy.
Option (C): Correct. I_0 approximately doubles every $10^{\circ}\mathrm{C}$ rise in temperature.
Option (D): Incorrect. The increase is exponential with temperature, not linear.

HIGH-YIELD EXAM TAKEAWAY: Reverse saturation current doubles approximately every $10^{\circ}\mathrm{C}$ rise in junction temperature.

Question 07 1.4 Semiconductor Devices: BJT Biasing

For a Bipolar Junction Transistor (BJT) to operate stably in its Active (linear amplification) region, what must be the respective biasing conditions of its two junctions?

✓ Correct Answer: **(B) Emitter-Base junction must be forward biased, and Collector-Base junction must be reverse biased**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The operating regions of a BJT are defined by the bias states of its two P-N junctions:
1. **Active Region (Amplification)**: Emitter-Base (EB) junction is **forward biased** (injecting majority carriers into the base), and Collector-Base (CB) junction is **reverse biased** (sweeping minority carriers into the collector).
2. **Saturation Region (ON switch)**: Both EB and CB junctions are **forward biased**.
3. **Cut-off Region (OFF switch)**: Both EB and CB junctions are **reverse biased**.
4. **Inverted Active Region**: EB reverse biased, CB forward biased.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. This biases the BJT into the Saturation region.
Option (B): Correct. Active region requires EB junction forward biased and CB junction reverse biased.
Option (C): Incorrect. This corresponds to the inverted active mode with poor current gain.
Option (D): Incorrect. This biases the BJT into the Cut-off region.

HIGH-YIELD EXAM TAKEAWAY: Active mode $\Longrightarrow$ EB forward biased + CB reverse biased ($I_C = \beta I_B$).

Question 08 1.4 Semiconductor Devices: MOSFET & CMOS

Why are MOSFETs overwhelmingly preferred over BJTs in high-density integrated circuits such as modern microprocessors and semiconductor memory?

✓ Correct Answer: **(B) MOSFETs have extremely high input impedance with negligible static gate current and occupy significantly smaller silicon chip area**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

MOSFETs (Metal-Oxide-Semiconductor Field-Effect Transistors) provide critical advantages in VLSI:
1. **Extremely High Input Impedance**: The gate is isolated by a thin layer of silicon dioxide (SiO_2), resulting in input resistance of $10^{12} - 10^{14}\,\Omega$ and practically zero DC gate current.
2. **Low Static Power Consumption**: In CMOS circuits, static power dissipation occurs only due to tiny leakage currents because one transistor of the complementary pair is always OFF.
3. **Superior Packing Density**: MOSFETs are compact and self-isolating, requiring far less silicon substrate area per transistor than BJTs without requiring deep isolation wells.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. MOSFETs require almost zero static gate drive current, unlike BJTs which require steady base current.

Option (B): Correct. High input impedance, negligible static power dissipation, and small physical size make MOSFETs ideal for VLSI.

Option (C): Incorrect. MOSFETs are unipolar majority-carrier devices, not minority-carrier devices.

Option (D): Incorrect. These are limitations, not reasons for preferring MOSFETs.

HIGH-YIELD EXAM TAKEAWAY: MOSFETs are unipolar, voltage-controlled devices with high input impedance and high packing density in CMOS VLSI.

Question 09 1.5 Signal Generator: Oscillators

According to the Barkhausen criterion, what are the two essential conditions required for an electronic amplifier feedback circuit to sustain continuous sinusoidal oscillations?

✓ Correct Answer: **(B) Loop gain magnitude $|A\beta| = 1$ and net phase shift around the feedback loop must be 0° or an integer multiple of 360°**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

For an amplifier with feedback transfer function $A_f = \frac{A}{1-A\beta}$ to sustain stable sinusoidal oscillations without an external input:
1. The magnitude of the loop gain must equal unity: $|A\beta| = 1$ (in practical start-up, $|A\beta| > 1$ initially, stabilizing to 1 via non-linear amplitude limiting).
2. The total phase shift around the closed feedback loop must equal 0° or 360° (positive feedback: $\angle A\beta = 2\pi n, n \in \mathbb{Z}$).
This ensures the feedback signal reinforces the original signal in phase with undiminished amplitude.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $|A\beta| = 0$ extinguishes the signal completely.

Option (B): Correct. Barkhausen criterion requires $|A\beta| = 1$ and loop phase shift $\angle A\beta = 0^\circ$ or 360°.

Option (C): Incorrect. 180° phase shift produces negative feedback, which stabilizes amplification rather than generating oscillation.

Option (D): Incorrect. Without feedback ($\beta = 0$), oscillation cannot occur.

HIGH-YIELD EXAM TAKEAWAY: Barkhausen Criterion for sustained oscillation: $|A\beta| = 1$ and $\angle A\beta = 0^\circ$ (or 360°).

Question 10 1.6 Amplifiers: Operational Amplifiers

In an operational amplifier (Op-Amp), the Common-Mode Rejection Ratio (CMRR) is formally defined as the ratio of:

✓ **Correct Answer: (A) Differential gain (A_d) to common-mode gain (A_{cm})**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Common-Mode Rejection Ratio (CMRR) measures an operational amplifier's ability to reject unwanted common-mode input signals (such as electrical noise, 50 Hz hum) while amplifying the desired differential signal:

$$\text{CMRR} = \left| \frac{A_d}{A_{cm}} \right|$$

In decibels (dB):

$$\text{CMRR}_{\text{dB}} = 20 \log_{10} \left| \frac{A_d}{A_{cm}} \right|$$

For an ideal Op-Amp, common-mode gain $A_{cm} = 0$, giving an infinite CMRR ($\text{CMRR} = \infty$). Practical IC op-amps (e.g. 741) achieve CMRR ≥ 90 dB.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. CMRR is the ratio of differential voltage gain to common-mode voltage gain (A_d/A_{cm}).

Option (B): Incorrect. This is the inverse of CMRR.

Option (C): Incorrect. This relates to input offset parameters, not common-mode rejection.

Option (D): Incorrect. Slew rate and bandwidth govern frequency and dynamic response limits.

HIGH-YIELD EXAM TAKEAWAY: $\text{CMRR} = \left| \frac{A_d}{A_{cm}} \right|$. Ideal Op-Amp: $A_d = \infty$, $A_{cm} = 0 \implies \text{CMRR} = \infty$.

SECTION II: DIGITAL LOGIC AND MICROPROCESSOR

Solutions Q.11 – Q.20

Question 11

2.1 Digital Logic: Number Systems & 2's Complement

In an 8-bit signed two's complement binary number system, what is the valid numerical range of representable decimal values, and what decimal value does the bit pattern `1111 0110` represent?

✓ Correct Answer: **(A) -128 to +127; the pattern represents -10**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

1. For an n-bit signed two's complement system, the representable integer range is:

$$-[2^{n-1}] \text{ to } +[2^{n-1}-1]$$

For $n = 8$ bits: -2^7 to $+(2^7-1) = -128$ to $+127$.

2. To find the decimal value of `1111 0110`:

- The Most Significant Bit (MSB) is 1, indicating a negative number.
- Invert all bits: `0000 1001` (binary 9).
- Add 1: `0000 1001 + 1 = 0000 1010` (decimal 10).
- Therefore, `1111 0110` represents **-10**.

Alternatively: $-128 \times 1 + 64 + 32 + 16 + 0 + 4 + 2 + 0 = -128 + 118 = -10$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. The 8-bit two's complement range is -128 to $+127$, and `1111 0110` equals -10.

Option (B): Incorrect. The range bounds are reversed and magnitude is miscalculated.

Option (C): Incorrect. This represents the unsigned 8-bit range (0 to 255).

Option (D): Incorrect. Arithmetic error in finding the 2's complement magnitude.

HIGH-YIELD EXAM TAKEAWAY: n-bit 2's complement range: -2^{n-1} to $+2^{n-1}-1$. Unsigned range: 0 to 2^n-1.

Question 12 2.1 Boolean Algebra & K-Maps

What is the minimal Sum-of-Products (SOP) Boolean expression for the function $F(A, B, C) = \sum m(1, 3, 5, 7)$?

✓ **Correct Answer: (C) C**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Examine the minterms in binary:
- $m(1) = 001$ ($A'B'C$)
- $m(3) = 011$ ($A'BC$)
- $m(5) = 101$ ($AB'C$)
- $m(7) = 111$ (ABC)

Notice that in all 4 minterms, variable C is strictly 1, while variables A and B take on all four possible combinations ($00, 01, 10, 11$).

On a 3-variable Karnaugh map, these 4 minterms form a contiguous quad of 4 adjacent cells. Eliminating variables A and B whose values change across the group yields:

$$F = C$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $A = 1$ for minterms 4, 5, 6, 7.

Option (B): Incorrect. $B = 1$ for minterms 2, 3, 6, 7.

Option (C): Correct. All minterms with odd decimal indices have $C = 1$ (least significant bit is 1).

Option (D): Incorrect. This is not the minimal prime implicant.

HIGH-YIELD EXAM TAKEAWAY: Minterms $(1, 3, 5, 7)$ in a 3-variable system reduce to a single literal: $F = C$.

Question 13 2.2 Combinational Circuits: Multiplexers

How many 4-to-1 multiplexers are required to construct a single 16-to-1 multiplexer without using any additional external logic gates?

✓ **Correct Answer: (B) 5**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

To build a 16-to-1 multiplexer using 4-to-1 multiplexers:

1. **First Stage**: To accept 16 primary inputs, four 4-to-1 multiplexers are needed ($4 \times 4 = 16$ inputs). They are controlled by the two lower select lines (S_1, S_0).

2. **Second Stage**: The 4 single-bit outputs from the first stage must be selected down to 1 final output. This requires one additional 4-to-1 multiplexer controlled by the two upper select lines (S_3, S_2).

Total number of 4-to-1 multiplexers required $= 4 + 1 = 5$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 4 multiplexers handle the 16 inputs but leave 4 separate outputs without selecting between them.

Option (B): Correct. 4 multiplexers in the first layer + 1 multiplexer in the second layer = 5 multiplexers.

Option (C): Incorrect. 8 is double the required number.

Option (D): Incorrect. 16 is the number of primary data inputs, not multiplexer ICs.

HIGH-YIELD EXAM TAKEAWAY: Tree multiplexing formula: $\lceil 16/4 \rceil + \lceil 4/4 \rceil = 4 + 1 = 5$ multiplexers.

Question 14 2.2 Combinational Circuits: Encoders

In a 4-to-2 Priority Encoder where input D_3 has the highest priority and D_0 has the lowest priority, what is the output code (Y_1Y_0) when inputs D_2 and D_1 are simultaneously active HIGH (with $D_3 = 0$ and $D_0 = 1$)?

✓ **Correct Answer: (C) 10**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In a priority encoder, if two or more inputs are active simultaneously, the output code corresponds to the input with the highest designated priority.
Here the input vector is $(D_3, D_2, D_1, D_0) = (0, 1, 1, 1)$:
- $D_3 = 0$ (inactive)
- $D_2 = 1$ (active)
- $D_1 = 1$ (active)
- $D_0 = 1$ (active)

Among the active inputs (D_2, D_1, D_0), D_2 has the highest priority. The binary encoded representation of decimal index 2 is:

$$Y_1Y_0 = 10$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 00 corresponds to D_0 having highest active priority.

Option (B): Incorrect. 01 corresponds to D_1 having highest active priority.

Option (C): Correct. D_2 is the highest active input, producing binary code 10.

Option (D): Incorrect. 11 corresponds to D_3 being active.

HIGH-YIELD EXAM TAKEAWAY: Priority encoders always encode the active input with the highest priority index ($D_2 \implies 10_2$).

Question 15 2.3 Sequential Logic: Flip-Flops

Under what specific condition does the 'Race-Around Condition' occur in a level-triggered JK flip-flop?

✓ **Correct Answer: (B) When $J = 1, K = 1$ and the clock pulse width exceeds the flip-flop propagation delay ($t_w > t_{pd}$)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The race-around condition occurs in level-triggered JK flip-flops when:
1. Both inputs are HIGH ($J = 1, K = 1$), which puts the flip-flop in toggle mode ($Q_{next} = Q'$).
2. The clock pulse width (t_w) is greater than the internal gate propagation delay (t_{pd}) of the flip-flop ($t_w > t_{pd}$).

When the clock is HIGH, the output toggles. Since the clock is still HIGH after time t_{pd}, the new output feeds back to the input gates and causes another toggle. The output oscillates unpredictably between 0 and 1 during the duration of a single clock pulse.
It is resolved by using **Master-Slave JK flip-flops** or **edge-triggered flip-flops**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $J = K = 0$ is the memory/hold state where no toggling occurs.

Option (B): Correct. $J = K = 1$ and $t_w > t_{pd}$ causes repeated toggling during a single clock pulse.

Option (C): Incorrect. $J = 1, K = 0$ sets the flip-flop to 1 without toggling.

Option (D): Incorrect. $J = 0, K = 1$ resets the flip-flop to 0.

HIGH-YIELD EXAM TAKEAWAY: Race-around condition occurs when $J = K = 1$ and $t_w > t_{pd}$. Solution: Edge-triggering or Master-Slave architecture.

Question 16 2.3 Sequential Logic: Counters

What is the total number of distinct operating states (modulus) for a 4-bit Johnson (twisted-ring) counter and a 4-bit standard Ring counter, respectively?

✓ Correct Answer: **(B) 8 and 4**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

For an n-bit shift-register-based counter:

1. **Standard Ring Counter**: Formed by circulating a single '1' (or '0') in a closed loop (ar{Q}_n not inverted; output Q_n feeds into D_1). A 4-bit ring counter has states `1000`, `0100`, `0010`, `0001`:

$$\text{Modulus} = n = 4$$

2. **Johnson (Twisted-Ring) Counter**: Formed by connecting the inverted output (ar{Q}_n) of the last flip-flop to the input (D_1) of the first flip-flop. This produces $2n$ states:

$$\text{Modulus} = 2n = 2 \times 4 = 8$$

(States: `0000`, `1000`, `1100`, `1110`, `1111`, `0111`, `0011`, `0001`).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 16 is the modulus of a full 4-bit binary counter (2^n).

Option (B): Correct. Johnson counter modulus is $2n = 8$; Ring counter modulus is $n = 4$.

Option (C): Incorrect. Ring counter has 4 states, not 16.

Option (D): Incorrect. The numbers are reversed.

HIGH-YIELD EXAM TAKEAWAY: Modulus of n-bit Ring Counter $= n$. Modulus of n-bit Johnson Counter $= 2n$.

Question 17 2.4 Microprocessor: 8085 Architecture

In the Intel 8085 microprocessor, how is the lower-order multiplexed Address/Data bus ($AD_0 - AD_7$) demultiplexed to obtain separate address lines ($A_0 - A_7$) and data lines ($D_0 - D_7$)?

✓ Correct Answer: **(B) By using the Address Latch Enable (ALE) signal to latch the lower address byte into an external octal latch (74LS373)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In the 8085 microprocessor, pins 12–19 are time-multiplexed as $AD_0 - AD_7$ to minimize pin count (40-pin DIP package):

- During the T_1 clock state of every machine cycle, these lines carry the lower 8 bits of the memory address ($A_0 - A_7$), and the 8085 asserts **ALE (Address Latch Enable)** HIGH.
- The trailing (falling) edge of the ALE pulse latches $A_0 - A_7$ into an external transparent latch (IC 74LS373).
- During T_2, T_3, T_4 states, ALE goes LOW and the bus lines are repurposed to transmit or receive bidirectional 8-bit data ($D_0 - D_7$).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 74LS245 buffers data directions; it does not latch addresses using ALE.

Option (B): Correct. ALE strobing the 74LS373 latch separates lower address bits from data.

Option (C): Incorrect. $IO/\overline{M}$ distinguishes memory from I/O cycles, not demultiplexing.

Option (D): Incorrect. READY synchronizes slow memory/peripherals.

HIGH-YIELD EXAM TAKEAWAY: 8085 Address/Data demultiplexing: ALE falling edge latches $A_0 - A_7$ via 74LS373.

Question 18 2.4 Microprocessor: Addressing Modes

The Intel 8085 assembly language instruction `LDAX B` loads the accumulator with data from the memory location specified by the BC register pair. Which addressing mode does this instruction employ?

✓ **Correct Answer: (C) Register Indirect addressing mode**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Addressing modes in 8085:
- **Direct Addressing**: The 16-bit operand address is explicitly included in the instruction (e.g., `LDA 2050H`).
- **Immediate Addressing**: The 8-bit or 16-bit data itself is specified in the instruction (e.g., `MVI A, 45H`, `LXI B, 1000H`).
- **Register Addressing**: Operands reside in internal registers (e.g., `MOV A, B`).
- **Register Indirect Addressing**: The instruction specifies a register pair that holds the 16-bit memory address where the operand is stored (e.g., `LDAX B`, `STAX D`, `MOV M, A`). Because `LDAX B` accesses memory via the address in `BC`, it is **Register Indirect**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Direct addressing explicitly writes the 16-bit address in bytes 2 and 3.
Option (B): Incorrect. Immediate addressing supplies the constant data value inside the instruction.
Option (C): Correct. The register pair BC serves as an indirect pointer to the target memory byte.
Option (D): Incorrect. Implied addressing operates on a fixed register without operands (e.g., `CMA`, `RAL`).

HIGH-YIELD EXAM TAKEAWAY: `LDAX rp` and instructions referencing memory through `M` (HL pair) use Register Indirect addressing.

Question 19 2.5 Microprocessor System: Peripheral Interfacing (8255 PPI)

Which operating mode of the Intel 8255 Programmable Peripheral Interface (PPI) supports bidirectional 8-bit bus data transfer on Port A with handshake signals provided by Port C?

✓ **Correct Answer: (C) Mode 2 (Strobed Bidirectional Bus I/O)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The Intel 8255 PPI operates in two main modes: BSR mode (for Port C bit set/reset) and I/O mode. The I/O mode has three sub-modes:
1. **Mode 0 (Basic I/O)**: Simple unconditional I/O without handshaking (Ports A, B, C).
2. **Mode 1 (Strobed I/O)**: Unidirectional input or output with handshake control signals on Port C (supported by Port A and Port B).
3. **Mode 2 (Strobed Bidirectional Bus I/O)**: Provides an 8-bit bidirectional data bus on **Port A only**, utilizing 5 control/handshake pins of Port C ($PC_3 - PC_7$). Port B can simultaneously operate in Mode 0 or Mode 1.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Mode 0 does not support handshaking or bidirectional bus transfer.
Option (B): Incorrect. Mode 1 is unidirectional strobed I/O.
Option (C): Correct. Mode 2 is specifically designed for strobed bidirectional bus communication on Port A.
Option (D): Incorrect. BSR mode only sets or resets individual pins of Port C.

HIGH-YIELD EXAM TAKEAWAY: 8255 Modes: Mode 0 = Basic I/O, Mode 1 = Strobed Unidirectional I/O, Mode 2 = Strobed Bidirectional I/O (Port A only).

Question 20 2.6 Interrupt Operations: 8085 Interrupts

Which hardware interrupt in the Intel 8085 microprocessor is non-maskable, holds the highest priority, and automatically transfers execution to vector memory address `0024H`?

✓ **Correct Answer: (D) TRAP**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Hardware interrupts in the 8085 in order of priority:
1. **TRAP (RST 4.5)**: Highest priority, **Non-Maskable** (cannot be disabled by `DI` instruction or `SIM` mask), edge-and-level triggered. Vector address $= 4.5 \times 8 = 36 = \mathbf{0024H}$.
2. **RST 7.5**: Maskable, positive edge-triggered. Vector $= 7.5 \times 8 = 60 = \mathbf{003CH}$.
3. **RST 6.5**: Maskable, level-triggered. Vector $= 6.5 \times 8 = 52 = \mathbf{0034H}$.
4. **RST 5.5**: Maskable, level-triggered. Vector $= 5.5 \times 8 = 44 = \mathbf{002CH}$.
5. **INTR**: Lowest priority, maskable, non-vectored (requires external device to supply `RST n` or `CALL` opcode).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. INTR is non-vectored, maskable, and lowest priority.
Option (B): Incorrect. RST 7.5 is maskable and vectors to 003CH.
Option (C): Incorrect. RST 6.5 is maskable and vectors to 0034H.
Option (D): Correct. TRAP is non-maskable, highest priority, and vectors to 0024H.

HIGH-YIELD EXAM TAKEAWAY: TRAP vector address: $4.5 \times 8 = 36_{10} = 0024\text{H}$. It is non-maskable and edge-and-level sensitive.

SECTION III: PROGRAMMING LANGUAGE AND ITS APPLICATIONS

Solutions Q.21 – Q.30

Question 21

3.1 C Programming: Operators & Short-Circuit Evaluation

What is the exact terminal output generated by the following C program segment?

```
int a = 0, b = 5;
if (a++ && ++b) {
    b += 2;
}
printf("%d %d", a, b);
```

✓ **Correct Answer: (A) 1 5**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Detailed step-by-step trace:
1. The `if` conditional expression is `(a++ && ++b)`.
2. The left operand is evaluated first: `a++`. The post-increment operator yields the current value of `a` (0) for the condition, and schedules `a` to be incremented to 1.
3. In C, logical AND (&&) employs **short-circuit evaluation**: because the left operand is false (0), the truth value of the entire expression is determined to be false.
4. Therefore, the right operand `++b` is **never executed**.
5. The side effect of `a++` completes at the sequence point, leaving $a = 1$.
6. The body of the `if` statement is bypassed because the condition evaluated to false (0).
7. `b` remains untouched at 5. The `printf` prints `1 5`.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. `a` is incremented to 1; short-circuit evaluation prevents `++b` from executing.
Option (B): Incorrect. Assumes `++b` executed, ignoring short-circuit evaluation rules.
Option (C): Incorrect. Fails to record the post-increment of `a`.
Option (D): Incorrect. Assumes the `if` body executed.

HIGH-YIELD EXAM TAKEAWAY: Short-circuit evaluation: For `&&`, if the left operand is 0, the right operand is never evaluated.

Question 22 3.2 Pointers, Structures & Files in C

Given the C declarations:

```
int arr[] = {10, 20, 30, 40, 50};
int *p = arr;
```

What is the evaluated integer result of the expression `*p++ + *(++p)`?

✓ Correct Answer: **(B) 40**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Step-by-step evaluation:
1. `p` initially points to `arr[0]` (value 10).
2. First term `*p++`:
- Due to operator precedence, postfix `++` has higher precedence than `*`.
- However, the post-increment returns the original pointer `p` (pointing to `arr[0]`), which is dereferenced to yield **10**.
- After this evaluation, pointer `p` advances to `arr[1]` (holding 20).
3. Second term `*(++p)`:
- The prefix `++p` immediately increments pointer `p` from `arr[1]` to `arr[2]` (holding 30).
- Dereferencing `*(++p)` yields **30**.
4. Sum $= 10 + 30 = \mathbf{40}$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Miscalculates the second dereference as `arr[1]` $(10 + 20 = 30)$.

Option (B): Correct. First term evaluates to 10 (advancing pointer to index 1), second term pre-increments to index 2 (value 30), total = 40.

Option (C): Incorrect. Assumes indices 1 and 2 $(20 + 30)$.

Option (D): Incorrect. Assumes indices 1 and 3.

HIGH-YIELD EXAM TAKEAWAY: `*p++` dereferences original pointer then increments; `*(++p)` increments pointer first then dereferences.

Question 23 3.2 Structures vs Unions: Memory Allocation

In C on a standard 64-bit architecture with 8-byte boundary alignment, what is `sizeof(union Sample)` for the following definition?

```
union Sample {
    int i;        /* 4 bytes */
    char str[21]; /* 21 bytes */
    double d;     /* 8 bytes */
};
```

✓ **Correct Answer: (B) 24 bytes**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Memory allocation rules for C unions:

1. In a union, all members share the same overlapping memory space. The total size must be at least large enough to hold its largest member.
2. Largest member: `char str[21]` requires 21 bytes.
3. **Alignment requirement**: The union must satisfy the strictest alignment constraint among all its members. The member `double d` requires an 8-byte boundary alignment.
4. Therefore, the union size must be padded up to the nearest multiple of 8 that is greater than or equal to 21 bytes:

$$\text{Nearest multiple of } 8 \geq 21 \implies 8 \times 3 = 24 \text{ bytes}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Sum of all members ($4 + 21 + 8 = 33$) corresponds to a `struct`, not a `union`.

Option (B): Correct. Largest element is 21 bytes, padded to the next 8-byte boundary = 24 bytes.

Option (C): Incorrect. 21 bytes ignores mandatory struct/union alignment padding.

Option (D): Incorrect. 8 bytes cannot hold the 21-byte character array.

HIGH-YIELD EXAM TAKEAWAY: Union size = size of largest member rounded up to a multiple of the largest member's alignment requirement.

Question 24 3.2 Dynamic Memory Allocation & Pointers

What is a 'dangling pointer' in C and C++ programming?

✓ Correct Answer: **(B) A pointer pointing to a memory location that has already been deallocated using `free()` or `delete`**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

A **dangling pointer** arises when the memory block pointed to by the pointer is deallocated (freed) while the pointer itself retains the memory address:

```
int *p = (int*)malloc(sizeof(int));
free(p); // Memory is deallocated, but 'p' still holds the address!
// 'p' is now a DANGLING POINTER.
```

Dereferencing or writing through a dangling pointer causes undefined behavior, security vulnerabilities, or segmentation faults. Best practice dictates immediately reassigning `p = NULL;` after deallocation.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. An uninitialized pointer is called a **wild pointer**.

Option (B): Correct. A pointer pointing to freed/deallocated memory is a dangling pointer.

Option (C): Incorrect. A null pointer is a safe, valid sentinel state.

Option (D): Incorrect. A `void*` is a generic pointer.

HIGH-YIELD EXAM TAKEAWAY: Dangling pointer = points to deallocated heap or expired stack memory. Fix: set pointer to `NULL` after freeing.

Question 25 3.3 C++ Classes & Objects: this Pointer

Which of the following statements regarding the `this` pointer in C++ is strictly TRUE?

✓ Correct Answer: **(B) The `this` pointer is passed as a hidden constant pointer argument to all non-static member functions, pointing to the invoking object instance**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Properties of the `this` pointer in C++:

1. It is a hidden, implicit parameter passed automatically to all non-static member functions. For class `Widget`, its type is `Widget* const this` (a constant pointer to the calling object).
2. Inside a non-static member function, `this->member` explicitly refers to the invoking object's attributes.
3. It is **not available** in static member functions because static functions belong to the class rather than any individual instance.
4. It cannot be reassigned or modified (`this = ptr` is a compiler error).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Static functions have no `this` pointer.

Option (B): Correct. `this` is a hidden pointer to the invoking instance for non-static methods.

Option (C): Incorrect. `this` is an rvalue/constant pointer and cannot be assigned to.

Option (D): Incorrect. `this` exists for all non-static member invocations regardless of whether the object is stack or heap allocated.

HIGH-YIELD EXAM TAKEAWAY: `this` is passed implicitly to non-static member functions; it does not exist in static member functions.

Question 26 3.4 OOP Features: Inheritance & Constructors

In C++, when an instance of a derived class `Derived` that inherits from base class `Base` is instantiated and subsequently destroyed, what is the exact execution order of their constructors and destructors?

✓ **Correct Answer: (B) Base constructor → Derived constructor → Derived destructor → Base destructor**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In object-oriented C++:
1. **Construction Order**: Proceeds from base to derived. The base class sub-object must be completely initialized before the derived class constructor can execute its body, ensuring base members are valid: **Base Constructor → Derived Constructor**.
2. **Destruction Order**: Strictly the reverse of construction (LIFO order). The derived class destructor runs first to release derived resources, followed by the base class destructor: **Derived Destructor → Base Destructor**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Derived constructor cannot run before base is constructed.
Option (B): Correct. Construction: Base → Derived; Destruction: Derived → Base.
Option (C): Incorrect. Destructor order must be the exact reverse of construction.
Option (D): Incorrect. Construction starts with the most base class.

HIGH-YIELD EXAM TAKEAWAY: Constructor order: Base first, then Derived. Destructor order: Derived first, then Base (LIFO).

Question 27 3.4 OOP Features: Operator Overloading

Which of the following groups of C++ operators CANNOT be overloaded under any circumstances according to the ISO C++ standard?

✓ **Correct Answer: (C) ., .*, ::, ?:, sizeof**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The ISO C++ standard explicitly forbids overloading the following five operators:
1. Direct member access operator: `.`
2. Pointer-to-member operator: `.*`
3. Scope resolution operator: `::`
4. Ternary conditional operator: `?:`
5. Object size query operator: `sizeof`
(Also `typeid` cannot be overloaded).
All other operators listed in options A, B, and D can be overloaded as member functions or friend functions.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Arithmetic operators are standard overload candidates.
Option (B): Incorrect. Relational comparison operators can be overloaded.
Option (C): Correct. None of `.`, `.*`, `::`, `?:`, `sizeof` can be overloaded in C++.
Option (D): Incorrect. Subscript `[]`, call `()`, arrow `->`, and assignment `=` can be overloaded (must be non-static members).

HIGH-YIELD EXAM TAKEAWAY: Operators that CANNOT be overloaded: `.`, `.*`, `::`, `?:`, `sizeof`, and `typeid`.

Question 28 3.5 Pure Virtual Function & Dynamic Binding

How is runtime (late) polymorphism and dynamic dispatch implemented under the hood by C++ compilers for classes containing virtual functions?

✓ Correct Answer: **(B) Through a Virtual Method Table (VTABLE) of function pointers and a hidden virtual table pointer (`vptr`) embedded inside each object**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

C++ dynamic dispatch mechanism:
1. For every class containing at least one virtual function, the compiler creates a static array called the **VTABLE (Virtual Method Table)** holding function pointers to the virtual methods.
2. Every object of that class is allocated a hidden pointer called ** `vptr` ** (typically at offset 0 of the object layout) that points to its class's VTABLE.
3. When a virtual method is called via a base pointer (`basePtr->vfunc()`), the compiler emits code to dereference the object's `vptr` , index into the VTABLE, and execute the resolved function pointer dynamically at runtime.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Templates and inlining represent compile-time (static) polymorphism.
Option (B): Correct. VTABLE + `vptr` is the universal compiler implementation for virtual function dispatch.
Option (C): Incorrect. RTTI is used for `dynamic_cast` and `typeid` , not routine virtual function calls.
Option (D): Incorrect. Hash map lookups describe dynamic scripting languages (e.g. Python), not compiled C++.

HIGH-YIELD EXAM TAKEAWAY: Dynamic polymorphism in C++ is achieved via VTABLE (per class) and `vptr` (per object instance).

Question 29 3.6 Generic Programming: Standard Template Library (STL)

In the C++ Standard Template Library (STL), which underlying data structure is typically used to implement `std::map` , and what is its guaranteed worst-case search time complexity for n elements?

✓ Correct Answer: **(B) Red-Black Tree (Self-balancing BST) with $O(\log n)$ search time**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

STL Container implementations:
1. `std::map` stores sorted key-value pairs where keys are unique. The C++ standard mandates logarithmic time complexity for search, insertion, and deletion: **$O(\log n)$**.
2. To guarantee $O(\log n)$ worst-case performance, almost all major STL implementations (GCC libstdc++, LLVM libc++, MSVC) implement `std::map` and `std::set` using a **Red-Black Tree** (a height-balanced binary search tree).
3. By contrast, `std::unordered_map` is implemented using a **Hash Table**, offering $O(1)$ average search time but $O(n)$ worst-case time.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Hash tables back `std::unordered_map` , not `std::map` .
Option (B): Correct. `std::map` uses a self-balancing Red-Black Tree, guaranteeing $O(\log n)$ search/insert/delete.
Option (C): Incorrect. Doubly linked list backs `std::list` .
Option (D): Incorrect. Min-heap backs `std::priority_queue` .

HIGH-YIELD EXAM TAKEAWAY: `std::map` $\implies$ Red-Black Tree ($O(\log n)$ sorted). `std::unordered_map` $\implies$ Hash Table ($O(1)$ average).

Question 30 3.6 Exception Handling Constructs

In C++, what crucial safety mechanism occurs during exception propagation when an exception is thrown from a deeply nested function call?

✓ Correct Answer: **(B) Stack unwinding automatically invokes destructors in reverse order for all local automatic objects constructed within the exited scopes**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Stack unwinding is a fundamental runtime guarantee in C++:
When an exception is thrown, the runtime searches up the call stack for a matching `catch` block. As execution frames are unwound and exited:
- The destructors for all local (automatic) objects created within those scopes are invoked in strict reverse order of construction.
- This mechanism is the bedrock of the **RAII (Resource Acquisition Is Initialization)** idiom, ensuring that smart pointers, mutex locks, and file handles release their underlying resources automatically even when exceptions occur.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Only the throwing thread unwinds its stack.
Option (B): Correct. Stack unwinding guarantees that destructors of all local objects are called.
Option (C): Incorrect. Raw heap pointers are NOT automatically freed, leading to memory leaks without RAII smart pointers.
Option (D): Incorrect. C++ does not roll back global state during exceptions.

HIGH-YIELD EXAM TAKEAWAY: Stack unwinding calls destructors of all automatic objects in reverse order between `throw` and `catch`.

SECTION IV: COMPUTER ORGANIZATION AND EMBEDDED SYSTEM

Solutions Q.31 – Q.40

Question 31 4.1 CPU Control Unit Design: Hardwired vs Microprogrammed

Compared to a microprogrammed control unit, a hardwired control unit:

✓ Correct Answer: **(C) Executes instructions faster because control signals are generated directly by combinational logic circuits, but is difficult to modify**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Comparison of Control Unit architectures:
- **Hardwired Control Unit**: Control logic is implemented directly using combinational logic gates, state machine flip-flops, and decoders. Signals are generated at the speed of hardware logic, enabling very fast execution (essential for RISC). However, modifying the instruction set requires redesigning the physical hardware layout.
- **Microprogrammed Control Unit**: Control signals are generated by fetching microinstructions stored in a Control Read-Only Memory (CROM). It is highly flexible and easy to modify/upgrade (common in CISC), but slower due to memory access latency.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Hardwired units are faster, not slower, and are inflexible.
Option (B): Incorrect. Storing control words in ROM defines a microprogrammed control unit.
Option (C): Correct. Hardwired units achieve maximum execution speed via combinational hardware, at the cost of design inflexibility.
Option (D): Incorrect. Hardwired units have zero microinstruction overhead.

HIGH-YIELD EXAM TAKEAWAY: Hardwired = faster, combinational gates, inflexible (RISC). Microprogrammed = slower, ROM-based, flexible (CISC).

Question 32 4.1 Instruction Pipelining & Hazards

In an instruction pipeline, a Read-After-Write (RAW) data hazard occurs when:

✓ Correct Answer: **(B) An instruction requires the result of an earlier instruction that has not yet completed its write-back stage to the register file**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Pipeline Hazards classification:
1. **Data Hazards**: Arise from data dependencies between instructions:
- **RAW (Read-After-Write / True Dependency)**: Instruction J attempts to read source register R before Instruction I writes to R. If J reads prematurely, it gets stale data. (Resolved by forwarding/bypassing or stalling).
- **WAR (Write-After-Read / Anti-dependency)**: J writes before I reads.
- **WAW (Write-After-Write / Output dependency)**: J writes before I writes.
2. **Structural Hazards**: Hardware resource conflict (Option A).
3. **Control Hazards**: Branch/jump instruction altering instruction fetch flow (Option C).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. This describes a Structural Hazard.

Option (B): Correct. RAW is a true data dependency where an instruction reads a register before an earlier instruction writes to it.

Option (C): Incorrect. This describes a Control (Branch) Hazard.

Option (D): Incorrect. This describes a WAR (Write-After-Read) hazard.

HIGH-YIELD EXAM TAKEAWAY: RAW Hazard = True Data Dependency (Read before Write completes). Solution: Operand Forwarding or Stalls.

Question 33 4.2 Memory Hierarchy & Cache Design

A 32-bit byte-addressable computer system has a 64 KB 4-way set-associative cache with a block (line) size of 64 bytes. What are the bit widths of the Tag, Set Index, and Block Offset fields, respectively?

✓ **Correct Answer: (A) Tag = 18 bits, Index = 8 bits, Offset = 6 bits**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Derivation of cache address bit fields:
1. **Address width**: 32 bits.
2. **Block Offset bits**: Block size $= 64$ bytes $= 2^6$ bytes $\implies$ **6** bits.
3. **Total lines in cache**:

$$\text{Total Lines} = \frac{\text{Cache Size}}{\text{Block Size}} = \frac{64\ \text{KB}}{64\ \text{bytes}} = \frac{65536}{64} = 1024\ \text{lines}$$

4. **Number of Sets** (for 4-way set-associative):

$$\text{Number of Sets} = \frac{\text{Total Lines}}{\text{Associativity}} = \frac{1024}{4} = 256\ \text{sets} = 2^8\ \text{sets} \implies \mathbf{8}\ \text{bits for Set Index}$$

5. **Tag bits**:

$$\text{Tag} = \text{Total Address Bits} - (\text{Index} + \text{Offset}) = 32 - (8 + 6) = 32 - 14 = \mathbf{18}\ \text{bits}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Tag = 18 bits, Index = 8 bits, Offset = 6 bits.
Option (B): Incorrect. Index = 10 bits corresponds to a direct-mapped cache (1024 sets), not 4-way.
Option (C): Incorrect. Index = 6 bits corresponds to 16-way associativity.
Option (D): Incorrect. Calculation error in set partitioning.

HIGH-YIELD EXAM TAKEAWAY: Cache address breakdown: $\text{Offset} = \log_2(\text{block size})$, $\text{Index} = \log_2(\text{sets})$, $\text{Tag} = \text{Address} - (\text{Index} + \text{Offset})$.

Question 34 4.2 Memory System: Cache Write Policies

In a cache memory subsystem employing a **Write-Back policy with a 'dirty bit':**

✓ Correct Answer: **(B) Write operations modify only the cache line, and main memory is updated only when that modified cache line is replaced (evicted)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Cache Write Policies:
- **Write-Through**: Every write to cache immediately updates main memory as well. Simple, keeps memory consistent, but generates heavy bus traffic.
- **Write-Back (Copy-Back)**: CPU writes only to the cache line and sets a status flag called the **dirty bit** (or modified bit) to 1. Main memory is NOT updated during the write. When a dirty block is eventually evicted to make room for a new block, it is written back to main memory. This dramatically reduces memory bus traffic and write latency.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Updating main memory immediately on every write describes the **Write-Through** policy.
Option (B): Correct. Write-Back modifies cache and defers main memory update until eviction of a dirty block.
Option (C): Incorrect. Secondary storage (disk) is not involved in cache write policies.
Option (D): Incorrect. Write misses are handled via write-allocate or no-write-allocate without interrupts.

HIGH-YIELD EXAM TAKEAWAY: Write-Back writes only to cache and marks dirty bit; writes to main memory only upon block eviction.

Question 35 4.2 Virtual Memory & Address Translation

What is the primary architectural purpose of a Translation Lookaside Buffer (TLB) in a paged virtual memory system?

✓ Correct Answer: **(B) To act as a high-speed associative hardware cache for recently used virtual-to-physical page translation entries, avoiding slow main memory page table lookups**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In virtual memory systems, translating a virtual address to a physical address requires accessing page tables located in physical RAM. This means every memory reference requires at least two physical memory accesses (one for page table lookup, one for actual data).
The **Translation Lookaside Buffer (TLB)** is a small, fast, hardware associative cache (CAM) built into the Memory Management Unit (MMU):
- It stores recently used Virtual Page Number (VPN) to Physical Frame Number (PFN) translations.
- On a **TLB hit** ($> 95\%$ of accesses), translation finishes in < 1 clock cycle without touching RAM page tables.
- On a **TLB miss**, the MMU or OS accesses the page table in memory, loads the mapping into the TLB, and resumes.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Caching disk blocks is the role of the OS buffer cache.
Option (B): Correct. TLB caches virtual-to-physical address translations to accelerate memory access.
Option (C): Incorrect. Swap space buffers manage disk pages, not the TLB.
Option (D): Incorrect. TLB does not handle cryptographic keys.

HIGH-YIELD EXAM TAKEAWAY: TLB is an MMU hardware cache that stores virtual-to-physical page translations to prevent double memory accesses.

Question 36 4.3 Input-Output Organization & DMA

During a **Cycle Stealing** Direct Memory Access (DMA) transfer mode:

✓ Correct Answer: **(B) The DMA controller acquires the system bus to transfer exactly one data word/byte, then relinquishes bus control back to the CPU**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

DMA Data Transfer Modes:
1. **Cycle Stealing Mode**: The DMA controller requests the system bus, takes control for **one memory cycle** to transfer a single byte or word, and immediately relinquishes the bus back to the CPU. If the CPU does not need the bus at that moment, execution continues without stalling. This interleaves I/O and CPU execution with low latency.
2. **Burst Mode (Block Transfer)**: The DMA controller seizes the bus and transfers an entire contiguous block of data continuously before releasing the bus. CPU is locked out for the entire duration.
3. **Transparent Mode**: DMA transfers data only when the CPU is executing internal cycles and not using the bus.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Transferring an entire block continuously describes **Burst Mode**.
Option (B): Correct. Cycle stealing transfers one word per bus cycle and releases the bus.
Option (C): Incorrect. The CPU is not reset; it continues executing instructions that do not require the bus.
Option (D): Incorrect. Data bypassing the CPU registers directly to/from memory is the definition of DMA.

HIGH-YIELD EXAM TAKEAWAY: Cycle Stealing DMA transfers 1 word per bus cycle then releases bus; Burst DMA transfers entire blocks continuously.

Question 37 4.3 Multiprocessor Architectures: Flynn's Taxonomy

According to Flynn's taxonomy of computer architectures, a modern Graphics Processing Unit (GPU) executing identical shader kernel instructions across thousands of independent data points simultaneously is categorized as:

✓ Correct Answer: **(B) SIMD (Single Instruction, Multiple Data)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Flynn's Taxonomy categories:
- **SISD**: Traditional single-core uniprocessor executing one instruction stream on one data stream.
- **SIMD (Single Instruction, Multiple Data)**: A single control unit broadcasts one instruction to multiple processing elements (ALUs), each operating synchronously on different data elements (e.g., vector processors, GPU warps/wavefronts, AVX vector units).
- **MISD**: Multiple instructions operate on a single data stream (rare; fault-tolerant flight computers).
- **MIMD**: Multiple autonomous processors concurrently execute different instruction streams on different data streams (e.g., modern multi-core CPUs, distributed clusters).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. SISD has no parallel data execution.
Option (B): Correct. GPUs use SIMD (or SIMT - Single Instruction Multiple Threads) to process parallel data vectors.
Option (C): Incorrect. MISD executes multiple instructions on one data point.
Option (D): Incorrect. MIMD processors execute independent instruction streams, unlike synchronized GPU lockstep execution.

HIGH-YIELD EXAM TAKEAWAY: GPU / Vector processors = SIMD. Multi-core distributed systems = MIMD. Single-core traditional CPU = SISD.

Question 38 4.4 Embedded Systems: Hardware Architecture

Many embedded microcontrollers (such as AVR and PIC) are based on the **Harvard Architecture rather than the von Neumann architecture primarily because:**

✓ Correct Answer: **(B) Harvard architecture uses physically separate memories and buses for program instructions and data, enabling simultaneous instruction fetch and data read/write**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Architectural comparison:
- **von Neumann Architecture**: Instructions and data share the same physical memory space and the same shared system bus. Consequently, the CPU cannot read an instruction and read/write data simultaneously (the *von Neumann bottleneck*).
- **Harvard Architecture**: Uses physically distinct memory blocks and independent buses for program code (ROM/Flash) and data (RAM). The CPU can fetch the next instruction while concurrently reading or writing data operands in memory, achieving higher throughput and deterministic execution in embedded microcontrollers.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Sharing a single bus describes the von Neumann architecture.
Option (B): Correct. Separate physical memories and buses allow simultaneous instruction fetch and data access.
Option (C): Incorrect. An ALU is fundamental to all CPU architectures.
Option (D): Incorrect. All synchronous architectures require clock sources.

HIGH-YIELD EXAM TAKEAWAY: Harvard architecture has separate buses and memories for instructions and data, eliminating the von Neumann bottleneck.

Question 39 4.5 Real-Time Operating Systems: Task Scheduling

In a Real-Time Operating System (RTOS), 'Priority Inversion' occurs when a high-priority task is blocked waiting for a shared mutex held by a low-priority task, while a medium-priority task preempts the low-priority task. What standard protocol resolves this issue?

✓ Correct Answer: **(B) Priority Inheritance Protocol (PIP)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Priority Inversion scenario:
- Task L (Low priority) locks a shared resource (mutex M).
- Task H (High priority) becomes ready and requests M; it blocks because L holds M.
- Task M_{ed} (Medium priority), which does not need M, preempts L because $Priority(M_{ed}) > Priority(L)$.
- Task H is now indirectly blocked by Task M_{ed}, inverting the priority schedule!

Priority Inheritance Protocol (PIP) resolves this:
When Task H blocks on mutex M held by Task L, Task L temporarily inherits the higher priority of Task H. This prevents Task M_{ed} from preempting L. As soon as L releases mutex M, its priority reverts to normal, and H immediately runs.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Round-robin does not solve priority inversion and degrades real-time guarantees.
Option (B): Correct. Priority Inheritance Protocol temporarily raises the low-priority task's priority to prevent medium tasks from preempting it.
Option (C): Incorrect. FCFS ignores task priorities entirely.
Option (D): Incorrect. SJF is not a synchronization protocol.

HIGH-YIELD EXAM TAKEAWAY: Priority Inversion is solved by Priority Inheritance: task holding resource inherits priority of highest blocked task.

Question 40 4.6 Hardware Description Languages: VHDL

In VHDL digital hardware design, which construct defines the external input/output interface ports of a module, and which construct describes its internal logic behavior or structural interconnection?

✓ **Correct Answer: (B) `ENTITY` defines external ports; `ARCHITECTURE` defines internal logic**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

VHDL design units:
1. ** `ENTITY` **: Defines the external 'black box' view of the design, declaring the module name and its I/O interface using the `PORT` clause (port names, directions: `IN`, `OUT`, `INOUT`, and data types: `STD_LOGIC`, etc.).
2. ** `ARCHITECTURE` **: Defines the internal implementation of the entity. It describes how the circuit functions using concurrent signal assignments (`<=`), structural component instantiations, or sequential `PROCESS` blocks.
3. A single `ENTITY` can have multiple different `ARCHITECTURE` implementations (e.g. behavioral, dataflow, or structural).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. The roles are reversed.
Option (B): Correct. `ENTITY` specifies external I/O ports; `ARCHITECTURE` specifies internal logic/structure.
Option (C): Incorrect. `PACKAGE` contains shared types/constants; `CONFIGURATION` binds entity and architecture.
Option (D): Incorrect. `PROCESS` is an internal sequential block; `SIGNAL` represents internal wiring.

HIGH-YIELD EXAM TAKEAWAY: VHDL structure: `ENTITY` = external interface (I/O pins); `ARCHITECTURE` = internal logic implementation.

SECTION V: CONCEPT OF COMPUTER NETWORK AND NETWORK SECURITY SYSTEM

Solutions Q.41 – Q.50

Question 41 5.1 Networking Models: Protocol Data Units (PDUs)

What are the standard Protocol Data Unit (PDU) designations at the Transport, Network, and Data Link layers of the OSI model?

✓ **Correct Answer: (B) Segment, Packet, Frame**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

OSI Layer PDU designations during data encapsulation:
- **Layer 7, 6, 5 (Application, Presentation, Session)**: Data / Message
- **Layer 4 (Transport Layer)**: **Segment** (for TCP) or **Datagram** (for UDP)
- **Layer 3 (Network Layer)**: **Packet** (e.g., IP Packet)
- **Layer 2 (Data Link Layer)**: **Frame** (e.g., Ethernet Frame)
- **Layer 1 (Physical Layer)**: **Bits** (electrical, optical, or radio pulses)
Thus, Transport → Segment; Network → Packet; Data Link → Frame.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Reverses Transport and Network PDUs.
Option (B): Correct. Layer 4 = Segment, Layer 3 = Packet, Layer 2 = Frame.
Option (C): Incorrect. Reverses Data Link and Transport PDUs.
Option (D): Incorrect. Message is application layer; byte is a unit of storage.

HIGH-YIELD EXAM TAKEAWAY: PDU hierarchy: Application (Data) → Transport (Segment) → Network (Packet) → Data Link (Frame) → Physical (Bits).

Question 42 5.2 Data Link Layer: Error Detection (CRC)

In Cyclic Redundancy Check (CRC) error detection, if the generator polynomial is $G(x) = x^4 + x + 1$, how many bits of Frame Check Sequence (FCS / CRC remainder) will be appended to the transmitted data frame?

✓ **Correct Answer: (B) 4 bits**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Properties of CRC generator polynomials:
1. The generator polynomial is $G(x) = x^4 + x^1 + x^0$, which in binary is represented as `10011` (length 5 bits).
2. The degree of the generator polynomial (k) is defined by its highest power of x, which is **$k = 4$**.
3. In CRC calculation, k zeros are appended to the message before polynomial modulo-2 division.
4. The resulting remainder has a maximum degree of $k - 1 = 3$, meaning it fits into exactly **$k = 4$ bits**.
5. Therefore, exactly 4 checksum bits are appended as the FCS.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 5 bits is the number of coefficients in $G(x)$, not the degree.

Option (B): Correct. The number of CRC check bits appended equals the degree of $G(x)$, which is 4.

Option (C): Incorrect. 3 is the degree of the remainder polynomial ($k - 1$), but the remainder field width is $k = 4$ bits.

Option (D): Incorrect. 1 bit corresponds to a simple parity check, not a 4th-degree CRC.

HIGH-YIELD EXAM TAKEAWAY: Number of CRC bits appended = degree of generator polynomial $G(x)$ ($x^k \implies k$ bits).

Question 43 5.2 MAC Protocols: CSMA/CD Minimum Frame Size

In a 100 Mbps CSMA/CD Ethernet network with a maximum one-way propagation delay of $t_p = 25.6\,\mu s$, what is the theoretical minimum frame size required to guarantee that collisions are reliably detected?

✓ **Correct Answer: (D) 640 bytes**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

In CSMA/CD, a transmitting node must continue transmitting long enough to detect a collision from the farthest station. The transmission time (T_{tx}) must be at least twice the one-way propagation delay ($2 \times t_p$, the round-trip time RTT):

$$T_{tx} \geq 2 \times t_p = 2 \times 25.6\,\mu s = 51.2\,\mu s$$

Given data rate $R = 100\ \text{Mbps} = 100 \times 10^6\ \text{bits/s}$:

$$\text{Minimum Frame Size (bits)} = R \times T_{tx} = (100 \times 10^6) \times (51.2 \times 10^{-6}) = 5120\ \text{bits}$$

Converting to bytes:

$$\text{Minimum Frame Size} = \frac{5120\ \text{bits}}{8\ \text{bits/byte}} = \mathbf{640}\ \text{bytes}$$

(Note: In legacy 10 Mbps Ethernet with $t_p = 25.6\,\mu s$, minimum frame size is $\frac{10 \times 10^6 \times 51.2\mu s}{8} = 64$ bytes; at 100 Mbps, it scales by a factor of 10 to 640 bytes).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 64 bytes is the standard minimum frame size for 10 Mbps Ethernet.
Option (B): Incorrect. Arbitrary intermediate number.
Option (C): Incorrect. 512 bytes ignores the propagation round-trip calculation.
Option (D): Correct. $L_{min} = 2 \times t_p \times R = 2 \times 25.6\mu s \times 100\ \text{Mbps} = 5120\ \text{bits} = 640\ \text{bytes}$.

HIGH-YIELD EXAM TAKEAWAY: CSMA/CD collision detection rule: $T_{tx} \geq 2t_p \implies L_{min} \geq 2 \times t_p \times \text{Bandwidth}$.

Question 44 5.3 Network Layer: IPv4 Subnetting & CIDR

An enterprise network is allocated the IPv4 CIDR block `192.168.10.0/26`. What is its subnet mask, the number of assignable host IP addresses, and its directed broadcast address?

✓ Correct Answer: **(A) Subnet Mask: 255.255.255.192; Usable Hosts: 62; Broadcast: 192.168.10.63**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Calculation for `192.168.10.0/26`:
1. **Subnet Mask**: Prefix `/26` means 26 contiguous network bits:
`11111111.11111111.11111111.11000000` $= \mathbf{255.255.255.192}$.
2. **Host Bits (h)**: $32 - 26 = 6$ bits.
3. **Total IP addresses per subnet**: $2^6 = 64$.
4. **Usable Host IP addresses**: $2^h - 2 = 64 - 2 = \mathbf{62}$ hosts (excluding network ID and broadcast address).
5. **Range of Subnet**: `192.168.10.0` to `192.168.10.63`.
- Network Address: `192.168.10.0`
- First Usable Host: `192.168.10.1`
- Last Usable Host: `192.168.10.62`
- Directed Broadcast Address: **`192.168.10.63`**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Mask: 255.255.255.192, 62 usable hosts ($2^6 - 2$), broadcast: 192.168.10.63.
Option (B): Incorrect. Corresponds to a `/25` subnet.
Option (C): Incorrect. Corresponds to a `/27` subnet.
Option (D): Incorrect. Corresponds to a `/28` subnet.

HIGH-YIELD EXAM TAKEAWAY: Subnetting `/26`: Mask = 255.255.255.192, Block size = 64, Usable hosts = $64 - 2 = 62$.

Question 45 5.3 Network Layer: Routing Protocols

What primary problem causes the 'Count-to-Infinity' phenomenon in Distance Vector routing protocols (e.g., RIP), and which technique is used to combat it?

✓ Correct Answer: **(B) Slow propagation of link failure information leading to routing loops; mitigated by Split Horizon with Poison Reverse**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Distance Vector routing (based on the Bellman-Ford algorithm) suffers from the **Count-to-Infinity problem**:
- When a link fails, neighboring routers do not have a global network map. Router A updates its distance to unreachable network X via Router B, while Router B believes it can reach X via Router A. They increment their hop counts to each other in a loop until metric reaches infinity (16 in RIP).
- **Split Horizon**: A router never advertises a route back out the same interface through which it learned that route.
- **Poison Reverse**: When advertising a route back to the neighbor it learned it from, the router explicitly sets the metric to ∞ (16), immediately breaking mutual loops.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Flooding and Dijkstra belong to Link State protocols (OSPF).
Option (B): Correct. Count-to-Infinity is a Distance Vector loop problem solved by Split Horizon and Poison Reverse.
Option (C): Incorrect. Describes BGP exterior routing.
Option (D): Incorrect. STP prevents Layer 2 switching loops, not Layer 3 routing loops.

HIGH-YIELD EXAM TAKEAWAY: Count-to-Infinity in Distance Vector protocols is mitigated by Split Horizon with Poison Reverse and Hold-down timers.

Question 46 5.4 Transport Layer: TCP Connection Establishment

During the standard TCP Three-Way Handshake used to establish a reliable transport connection, what is the exact sequence of control flag packets exchanged between client and server?

✓ **Correct Answer: (B) Client sends `SYN` → Server sends `SYN+ACK` → Client sends `ACK`**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The TCP Three-Way Handshake steps:
1. **Step 1 (SYN)**: The client initiates connection by sending a TCP segment with the `SYN` flag set and an initial sequence number ISN_c (e.g. `SEQ = x`).
2. **Step 2 (SYN + ACK)**: The server responds with both `SYN` and `ACK` flags set. It acknowledges client's sequence number (`ACK = x + 1`) and sends its own initial sequence number (`SEQ = y`).
3. **Step 3 (ACK)**: The client replies with an `ACK` flag set (`ACK = y + 1`, `SEQ = x + 1`). Data payload may optionally be included. The connection is now in the `ESTABLISHED` state.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Connection cannot begin with an `ACK`.
Option (B): Correct. Step 1: SYN, Step 2: SYN+ACK, Step 3: ACK.
Option (C): Incorrect. `FIN` initiates TCP four-way connection termination.
Option (D): Incorrect. `RST` forcibly resets an invalid or rejected connection.

HIGH-YIELD EXAM TAKEAWAY: TCP 3-Way Handshake: Client `SYN` → Server `SYN–ACK` → Client `ACK`.

Question 47 5.4 Transport Layer: TCP Congestion Control

In TCP Reno congestion control, how does the congestion window size (`cwnd`) grow during the **Slow Start phase upon receiving acknowledgments?**

✓ **Correct Answer: (B) It doubles every Round Trip Time (RTT), exhibiting exponential growth until reaching `ssthresh`**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

TCP Reno Congestion Control phases:
1. **Slow Start**: Initial `cwnd` starts at 1 to 10 MSS. For every ACK received, `cwnd` increases by 1 MSS. Thus, for each RTT where all transmitted segments are acknowledged, `cwnd` doubles ($1 \to 2 \to 4 \to 8 \ldots$). This represents **exponential growth**.
2. **Congestion Avoidance**: Once `cwnd` reaches the slow-start threshold (`ssthresh`), TCP transitions to Additive Increase / Multiplicative Decrease (AIMD). `cwnd` increases **linearly** by 1 MSS per RTT.
3. **Packet Loss**: If a timeout occurs, `ssthresh` is set to $\text{cwnd}/2$, and `cwnd` resets to 1 MSS.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Linear increase (+1 MSS/RTT) occurs during **Congestion Avoidance**, not Slow Start.
Option (B): Correct. In Slow Start, `cwnd` increases exponentially, doubling every RTT.
Option (C): Incorrect. Window size is dynamic.
Option (D): Incorrect. Halving occurs upon packet loss (multiplicative decrease).

HIGH-YIELD EXAM TAKEAWAY: TCP Slow Start $\implies$ Exponential growth (2^k per RTT); Congestion Avoidance $\implies$ Linear growth (+1 MSS per RTT).

Question 48 5.5 Application Layer: DNS Resolution

What is the key functional difference between a **recursive DNS query and an **iterative** DNS query?**

✓ Correct Answer: **(B) In a recursive query, the queried DNS server assumes full responsibility to resolve the domain and return the final IP address; in an iterative query, the server returns the best referral IP address to the next authoritative server**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

DNS Query Resolution types:
- **Recursive Query**: The client (resolver) requests that the local DNS server resolve the name completely. The queried server takes on the burden of contacting root servers, TLD servers, and authoritative servers sequentially on the client's behalf, eventually returning either the final IP address or an error.
- **Iterative Query**: The client or local DNS server asks a remote DNS server for an answer. If the remote server does not hold the exact record, it does not query other servers; instead, it returns a **referral** (the IP address of the next DNS server down the hierarchy) for the querier to contact directly.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. The roles are reversed.

Option (B): Correct. Recursive = 'resolve it completely for me'; Iterative = 'give me the answer or tell me who to ask next'.

Option (C): Incorrect. Both query types typically use UDP port 53 (TCP is used for zone transfers or messages > 512 bytes).

Option (D): Incorrect. Forward and reverse lookups can both be recursive or iterative.

HIGH-YIELD EXAM TAKEAWAY: Recursive query: server finds final answer for client. Iterative query: server returns referral to next server.

Question 49 5.6 Network Security: Cryptography (RSA)

In the RSA public-key cryptographic algorithm, given two prime numbers $p = 7$ and $q = 11$, and a public encryption exponent $e = 13$, what is the modulus n and the private decryption key d?

✓ **Correct Answer: (A)** $n = 77, d = 37$

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

RSA key generation steps:
1. Calculate modulus n:

$$n = p \times q = 7 \times 11 = \mathbf{77}$$

2. Calculate Euler's totient function $\phi(n)$:

$$\phi(n) = (p-1)(q-1) = (7-1)(11-1) = 6 \times 10 = 60$$

3. Find private decryption key d such that:

$$d \times e \equiv 1 \pmod{\phi(n)} \implies 13 \times d \equiv 1 \pmod{60}$$

4. Using the Extended Euclidean Algorithm:
- $13 \times 1 = 13 \pmod{60}$
- $13 \times 37 = 481 = (8 \times 60) + 1 = 1 \pmod{60}$
Thus, $d = \mathbf{37}$.
(Public key $= (e, n) = (13, 77)$; Private key $= (d, n) = (37, 77)$).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. $n = 77$, and $d = 37$ since $(13 \times 37) = 481 = 8 \times 60 + 1$.

Option (B): Incorrect. $d = 13$ yields $13 \times 13 = 169 = 49 \pmod{60} \neq 1$.

Option (C): Incorrect. $n = p \times q = 77$, not 70.

Option (D): Incorrect. n must be 77, not 60.

HIGH-YIELD EXAM TAKEAWAY: RSA key relations: $n = p \cdot q$, $\phi(n) = (p-1)(q-1)$, and $d \cdot e \equiv 1 \pmod{\phi(n)}$.

Question 50 5.6 Network Security: IPsec Modes

In IPsec (IP Security), how does **Tunnel Mode fundamentally differ from **Transport Mode**?**

✓ **Correct Answer: (B) Tunnel Mode encrypts the entire original IP packet (header and payload) and encapsulates it inside a completely new outer IP header, typically used for site-to-site VPN gateways**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

IPsec Modes of Operation:
- **Transport Mode**:
- Encrypts and/or authenticates only the IP **payload** (data from Layer 4 upwards).
- The original IP header is preserved and visible.
- Used primarily for end-to-end communication between two individual hosts.
- **Tunnel Mode**:
- Encrypts and protects the **entire original IP packet** (both original IP header and payload).
- Appends a brand-new outer IP header that routes the packet between security gateways (VPN routers/firewalls).
- The original source and destination IP addresses are hidden from intermediate transit routers.
- Standard for Site-to-Site and Remote-Access VPNs.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Transport mode preserves the original IP header unencrypted.
Option (B): Correct. Tunnel mode encrypts the entire packet and adds a new outer IP header for gateway-to-gateway VPN.
Option (C): Incorrect. Both IPsec modes operate strictly at Layer 3 (Network Layer).
Option (D): Incorrect. Tunnel mode provides full ESP encryption and AH authentication.

HIGH-YIELD EXAM TAKEAWAY: Transport Mode = encrypts payload only (host-to-host). Tunnel Mode = encrypts entire packet + new IP header (VPN gateway).

SECTION VI: THEORY OF COMPUTATION AND COMPUTER GRAPHICS

Solutions Q.51 – Q.60

Question 51 6.1 Finite Automata: DFA State Minimization

What is the minimum number of states in a Deterministic Finite Automaton (DFA) that accepts the language over alphabet $\Sigma = \{0, 1\}$ consisting of all strings ending with the substring `01`?

✓ **Correct Answer: (B) 3**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

To design the minimal DFA for strings ending in `01`:
- State q_0 (Start state): Represents strings that do not end in `0` or `01` (e.g. empty string, or ending in `1`).
- State q_1: Represents strings whose current suffix is `0`.
- State q_2 (Accepting state): Represents strings whose current suffix is `01`.
Transitions:
- From q_0: on `0` $\rightarrow q_1$, on `1` $\rightarrow q_0$.
- From q_1: on `0` $\rightarrow q_1$ (self-loop, still ends in 0), on `1` $\rightarrow q_2$ (now ends in 01).
- From q_2: on `0` $\rightarrow q_1$ (suffix becomes 0), on `1` $\rightarrow q_0$ (suffix becomes 1).
All three states are pairwise distinguishable. Thus, the minimal DFA requires exactly **3 states**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 2 states cannot distinguish the suffix `0` from other non-accepting prefixes.
Option (B): Correct. Exactly 3 states (q_0, q_1, q_2) are needed.
Option (C): Incorrect. 4 states represents an unminimized NFA-to-DFA conversion without state merging.
Option (D): Incorrect. Excessive states.

HIGH-YIELD EXAM TAKEAWAY: For alphabet $\{0, 1\}$, a DFA recognizing strings ending in a pattern of length k requires $k + 1$ states (here $2 + 1 = 3$).

Question 52 6.1 Regular Expressions & Closure Properties

Regular languages are closed under which of the following complete sets of formal operations?

✓ **Correct Answer: (B) Union, Intersection, Complement, Concatenation, and Difference**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Closure properties of Regular Languages:
Regular languages are extremely robust and closed under all standard Boolean and algebraic operations:
1. **Union**: $L_1 \cup L_2$ (via NFA parallel branching)
2. **Concatenation**: $L_1 L_2$ (connecting final states of L_1 to start of L_2)
3. **Kleene Star**: L^* (looping final states back to start state)
4. **Complement**: $\bar{L} = \Sigma^* \setminus L$ (swapping accepting and non-accepting states in a complete DFA)
5. **Intersection**: $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$ (by De Morgan's laws or Cartesian product DFA)
6. **Difference**: $L_1 - L_2 = L_1 \cap \overline{L_2}$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. While true, it omits intersection, complement, and difference under which regular languages are also closed.
Option (B): Correct. Regular languages are closed under union, intersection, complement, concatenation, and difference.
Option (C): Incorrect. Regular languages are strictly closed under complement.
Option (D): Incorrect. Irrelevant classification.

HIGH-YIELD EXAM TAKEAWAY: Regular languages form a Boolean algebra: closed under Union, Intersection, Complement, Concatenation, Star, and Difference.

Question 53 6.1 Pumping Lemma for Regular Languages

Which of the following formal languages over $\Sigma = \{a, b\}$ is NON-REGULAR and cannot be recognized by any finite automaton?

✓ **Correct Answer: (C)** $L_3 = \{a^n b^n \mid n \geq 0\}$

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Why $L_3 = \{a^n b^n \mid n \geq 0\}$ is non-regular:
- A Finite Automaton has a strictly finite amount of internal memory (finite number of states).
- In L_3, to ensure that the number of b's matches the number of preceding a's, the machine must count an arbitrarily large number n.
- Since n is unbounded, a finite automaton would require infinitely many states to count n, which contradicts the definition of a *finite* automaton.
- Formally proven non-regular using the Pumping Lemma for regular languages. It is a Deterministic Context-Free Language recognized by a Pushdown Automaton using a stack.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Requires a simple 2-state DFA tracking odd/even parity.
Option (B): Incorrect. Expressible by regular expression $a^+ b^+$; recognized by a 3-state DFA.
Option (C): Correct. $a^n b^n$ requires infinite counting memory, proven non-regular by Pumping Lemma.
Option (D): Incorrect. Expressible by regular expression $a(a+b)^* a + a$; recognized by a simple DFA.

HIGH-YIELD EXAM TAKEAWAY: $L = \{a^n b^n \mid n \geq 0\}$ is non-regular because finite automata cannot count unbounded symbols without a stack.

Question 54 6.2 Context-Free Languages & Pushdown Automata

Which of the following statements regarding Pushdown Automata (PDA) is strictly TRUE?

✓ **Correct Answer: (B) NPDAs are strictly more powerful than DPDAs, because languages like even-length palindromes $\{ww^R\}$ cannot be accepted by any DPDA**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Automata hierarchy properties:
1. Unlike Finite Automata where DFA $\equiv$ NFA, **NPDAs are strictly more powerful than DPDAs** ($DCFL \subset CFL$).
2. The language of even-length palindromes $L = \{ww^R \mid w \in \{0,1\}^*\}$ requires the automaton to non-deterministically guess the exact midpoint of the string before popping characters from the stack. No deterministic PDA can recognize it.
3. A single-stack PDA cannot recognize $\{a^n b^n c^n\}$ because popping for b's empties the stack, leaving no way to verify the count of c's.
4. Adding a second stack makes a 2-PDA equivalent in computational power to a **Turing Machine**!

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Unlike finite automata, DPDA $\neq$ NPDA.
Option (B): Correct. NPDAs accept all CFLs; DPDAs accept only Deterministic CFLs.
Option (C): Incorrect. $\{a^n b^n c^n\}$ is context-sensitive and requires a Linear Bounded Automaton or Turing Machine.
Option (D): Incorrect. A 2-stack PDA has the full computational power of a universal Turing Machine.

HIGH-YIELD EXAM TAKEAWAY: NPDAs are strictly more expressive than DPDAs ($DCFL \subset CFL$). 2-stack PDA $\equiv$ Turing Machine.

Question 55 6.2 Context-Free Grammars: Normal Forms

In formal language theory, a context-free grammar is in Chomsky Normal Form (CNF) if all its production rules strictly follow which syntactic structure?

✓ **Correct Answer: (A) $A \to BC$ or $A \to a$ (where A, B, C are non-terminals and a is a single terminal)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Chomsky Normal Form (CNF) definition:
A Context-Free Grammar is in CNF if every production rule is of the form:
1. $A \to BC$ (a non-terminal deriving exactly two non-terminals), OR
2. $A \to a$ (a non-terminal deriving exactly one terminal symbol).
(Rule $S \to \epsilon$ is permitted if $\epsilon \in L$, provided S does not appear on the right-hand side of any production).
CNF is critical because parsing a string of length n takes exactly $2n - 1$ derivation steps, enabling the $O(n^3)$ CYK parsing algorithm.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. CNF allows only $A \to BC$ and $A \to a$.
Option (B): Incorrect. $A \to aB$ or $A \to a$ defines a **Regular (Right-Linear) Grammar**.
Option (C): Incorrect. Allows mixing of terminals and non-terminals, violating CNF.
Option (D): Incorrect. Three non-terminals on the RHS violates CNF.

HIGH-YIELD EXAM TAKEAWAY: Chomsky Normal Form: $A \to BC$ or $A \to a$. Derivation of string of length n requires exactly $2n - 1$ steps.

Question 56 6.3 Turing Machines & Computability

Alan Turing's seminal 1936 proof concerning the Halting Problem demonstrated that determining whether an arbitrary computer program will eventually halt or run forever is:

✓ Correct Answer: **(C) Undecidable (uncomputable), because no general algorithm can exist that solves the halting problem for all possible program-input pairs**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The Halting Problem and Undecidability:
- Turing proved that the problem of determining whether an arbitrary Turing Machine M will halt on input w is **undecidable**.
- Proof mechanism (Cantor's Diagonalization / Proof by Contradiction):
Assume a decider machine $H(P, I)$ exists that outputs 'halts' or 'loops'. Construct a paradoxical machine $D(P)$ that calls $H(P, P)$: if H says P halts, D enters an infinite loop; if H says P loops, D halts immediately. Passing D to itself ($D(D)$) causes a logical contradiction:
$D(D)$ halts $\iff$ $D(D)$ loops.
- Thus, no such general algorithm H can ever exist.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Halting problem is undecidable, not polynomial.
Option (B): Incorrect. It is not computable in any amount of time.
Option (C): Correct. The Halting Problem is formally undecidable.
Option (D): Incorrect. Nondeterministic Turing Machines are equivalent in decidability to deterministic ones.

HIGH-YIELD EXAM TAKEAWAY: The Halting Problem is undecidable. Proved by Alan Turing using proof by contradiction and diagonalization.

Question 57 6.4 Computer Graphics: Frame Buffer Architecture

A raster graphics display system has a resolution of 1280×1024 pixels and operates in 24-bit True Color mode. What is the minimum frame buffer memory required to store a single full-screen image?

✓ Correct Answer: **(B) 3.75 MB**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Frame Buffer memory calculation:
1. Total number of pixels $= 1280 \times 1024 = 1,310,720$ pixels.
2. Color depth $= 24$ bits per pixel $= 3$ bytes per pixel (8 bits each for Red, Green, Blue).
3. Total memory in bytes:

$$\text{Size (bytes)} = 1,310,720 \times 3 = 3,932,160 \text{ bytes}$$

4. Convert bytes to Megabytes (binary MB, where $1 \text{ MB} = 1024 \times 1024 = 1,048,576$ bytes):

$$\text{Size (MB)} = \frac{3,932,160}{1,048,576} = \mathbf{3.75} \text{ MB}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 1.25 MB corresponds to 8-bit color (1 byte per pixel).
Option (B): Correct. $1280 \times 1024 \times 3 \text{ bytes} / 1048576 = 3.75$ MB.
Option (C): Incorrect. Arithmetic scaling error.
Option (D): Incorrect. 31.45 is the number of mega-*bits*, not mega-*bytes*.

HIGH-YIELD EXAM TAKEAWAY: Frame Buffer Size $= \frac{\text{Width} \times \text{Height} \times \text{Color Depth (bits)}}{8 \times 1024 \times 1024}$ MB.

Question 58 6.4 Line Drawing Algorithms: Bresenham's vs DDA

What is the primary computational advantage of Bresenham's line drawing algorithm over the Digital Differential Analyzer (DDA) algorithm?

✓ Correct Answer: **(B) Bresenham's algorithm uses only incremental integer additions, subtractions, and bit shifts, completely avoiding floating-point arithmetic and rounding operations**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Comparison of line drawing algorithms:
- **DDA Algorithm**: Computes line coordinates using incremental floating-point step sizes ($m = \Delta y / \Delta x$). Every step requires floating-point addition and an explicit call to `round()` to select the nearest pixel. Floating-point operations and rounding are computationally expensive in hardware and graphics rasterizers.
- **Bresenham's Algorithm**: Formulates an integer decision variable (p_k) that tracks the sign of the error distance to candidate pixels. Calculating p_{k+1} requires only integer additions, subtractions, and multiplying by 2 (a single left bit-shift: `<< 1`). It is fast and natively implementable in hardware without an FPU.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Bresenham completely avoids floating-point operations.
Option (B): Correct. Bresenham's algorithm replaces floating-point math with pure integer additions and bit shifts.
Option (C): Incorrect. Both algorithms require initial endpoint coordinates.
Option (D): Incorrect. Bresenham draws straight lines and can be extended to circles using Cartesian midpoints.

HIGH-YIELD EXAM TAKEAWAY: Bresenham's line algorithm replaces floating-point math and rounding with pure integer arithmetic and bit shifts.

Question 59 6.5 2D Transformations: Homogeneous Coordinates

In 2D computer graphics using homogeneous coordinates, what is the 3×3 transformation matrix for a counterclockwise rotation by an angle θ about the coordinate origin?

✓ **Correct Answer: (A)**

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Mathematical derivation of 2D rotation:
Let a point (x, y) be rotated counterclockwise by angle θ around the origin to (x', y'):

$$x' = x\cos\theta - y\sin\theta$$

$$y' = x\sin\theta + y\cos\theta$$

In 2D homogeneous coordinates, points are represented as column vectors $[x, y, 1]^T$:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

This matrix representation enables composition of translations, rotations, and scaling via matrix multiplication.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Matrix row 1: $[\cos\theta, -\sin\theta, 0]$, row 2: $[\sin\theta, \cos\theta, 0]$, row 3: $[0, 0, 1]$.

Option (B): Incorrect. This represents a clockwise rotation by θ (or counterclockwise by $-\theta$).

Option (C): Incorrect. Matrix entries are placed in the translation columns.

Option (D): Incorrect. Transposed trigonometric entries.

HIGH-YIELD EXAM TAKEAWAY: 2D counterclockwise rotation matrix around origin: $R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

Question 60 6.5 2D Line Clipping: Cohen-Sutherland Algorithm

In the Cohen-Sutherland line clipping algorithm, a 4-bit region outcode (TBRL: Top, Bottom, Right, Left) is assigned to endpoints P_1 and P_2. A line segment is trivially REJECTED (guaranteed to lie entirely outside the clip window) if:

✓ **Correct Answer: (C)** $\text{Code}(P_1)\ \text{AND}\ \text{Code}(P_2) \neq 0000$

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Cohen-Sutherland Line Clipping acceptance/rejection tests:
1. **Trivial Acceptance**: Both endpoints lie entirely inside the clipping viewport window:

$$\text{Code}(P_1) = 0000\ \text{AND}\ \text{Code}(P_2) = 0000 \iff \text{Code}(P_1)\ \text{OR}\ \text{Code}(P_2) = 0000$$

The line is drawn completely without clipping.
2. **Trivial Rejection**: Both endpoints lie on the same outside side of the clipping boundary (e.g. both are above, both are below, both are left, or both are right):

$$\text{Code}(P_1)\ \text{AND}\ \text{Code}(P_2) \neq 0000$$

Because they share a common outside region bit, the entire line segment lies outside the window and can be discarded without calculating edge intersections.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. This is the condition for **trivial acceptance**.
Option (B): Incorrect. This is another formulation of trivial acceptance.
Option (C): Correct. Bitwise logical AND $\neq 0000$ guarantees the segment is trivially rejected.
Option (D): Incorrect. XOR does not indicate shared outside regions.

HIGH-YIELD EXAM TAKEAWAY: Cohen-Sutherland: Trivial Accept $\iff \text{Code}_1\ \text{OR}\ \text{Code}_2 = 0000$; Trivial Reject $\iff \text{Code}_1\ \text{AND}\ \text{Code}_2 \neq 0000$.

SECTION VII: DATA STRUCTURES, DATABASE SYSTEM AND OPERATING SYSTEM

Solutions Q.61 – Q.70

Question 61

7.1 Data Structures: Postfix Evaluation Using Stack

What is the evaluated numerical result of the postfix (Reverse Polish) expression `6 3 2 + * 10 - 4 /`?

✓ **Correct Answer: (B) 5**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Stack-based evaluation step by step:

1. Push `6`: Stack = `[6]`
2. Push `3`: Stack = `[6, 3]`
3. Push `2`: Stack = `[6, 3, 2]`
4. Encounter `+`: Pop `2`, pop `3`. Compute $3 + 2 = 5$. Push `5`. Stack = `[6, 5]`
5. Encounter `*`: Pop `5`, pop `6`. Compute $6 \times 5 = 30$. Push `30`. Stack = `[30]`
6. Push `10`: Stack = `[30, 10]`
7. Encounter `-`: Pop `10`, pop `30`. Compute $30 - 10 = 20$. Push `20`. Stack = `[20]`
8. Push `4`: Stack = `[20, 4]`
9. Encounter `/`: Pop `4`, pop `20`. Compute $20/4 = 5$. Push `5`. Stack = `[5]`
10. Final result $= \mathbf{5}$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Result of incorrect operand ordering during division.
Option (B): Correct. Sequential stack evaluation yields exactly 5.
Option (C): Incorrect. Evaluation arithmetic error.
Option (D): Incorrect. Forgetting to divide by 4.

HIGH-YIELD EXAM TAKEAWAY: In postfix evaluation: for operator $\otimes$, pop b, pop a, compute $a \otimes b$, push result.

Question 62 7.1 Balanced Trees: AVL Tree Rotations

In an AVL tree, what is the permissible range of the Balance Factor ($BF = h_{left} - h_{right}$) for every node, and which tree rotation is required when an insertion causes an imbalance in the left subtree of the right child of a node?

✓ **Correct Answer: (A) $BF \in \{-1, 0, +1\}$; Right-Left (RL) Double Rotation**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

AVL Tree Balancing properties:
1. An AVL tree is a self-balancing binary search tree where the Balance Factor of every node must strictly satisfy:

$$BF = \text{height}(\text{left subtree}) - \text{height}(\text{right subtree}) \in \{-1, 0, +1\}$$

2. When an insertion causes an imbalance ($|BF| \geq 2$):
- **LL Imbalance** (inserted into left subtree of left child): Fixed by **Single Right Rotation**.
- **RR Imbalance** (inserted into right subtree of right child): Fixed by **Single Left Rotation**.
- **LR Imbalance** (inserted into right subtree of left child): Fixed by **Left-Right (LR) Double Rotation**.
- **RL Imbalance** (inserted into left subtree of right child): Fixed by **Right-Left (RL) Double Rotation** (Right rotate on child, then Left rotate on root).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. $BF \in \{-1, 0, +1\}$, and insertion into the left subtree of a right child is an RL imbalance requiring an RL rotation.
Option (B): Incorrect. $BF = \pm 2$ is an illegal unbalanced state that triggers rebalancing.
Option (C): Incorrect. LR rotation fixes an insertion into the right subtree of a left child.
Option (D): Incorrect. -1 is a valid balance factor.

HIGH-YIELD EXAM TAKEAWAY: AVL tree balance factor: $\{-1, 0, +1\}$. RL imbalance is fixed by Right-Left double rotation.

Question 63 7.2 Sorting & Computational Complexity: QuickSort

What are the average-case and worst-case time complexities of standard QuickSort on an array of n elements, and under what condition does the worst-case time complexity occur when using the first element as the pivot?

✓ **Correct Answer: (B) Average $O(n \log n)$, Worst $O(n^2)$; when the array is already sorted or reverse-sorted**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

QuickSort Complexity Analysis:
- **Average-Case**: Partitioning splits the array roughly in half, leading to recurrence $T(n) = 2T(n/2) + O(n) = \mathbf{O(n \log n)}$.
- **Worst-Case**: Occurs when the pivot chosen is always the minimum or maximum element in the sub-array. If the first element is selected as pivot in an array that is **already sorted** or **reverse sorted**, one partition has 0 elements and the other has $n - 1$ elements:

$$T(n) = T(n-1) + O(n) = O(n^2)$$

(Mitigated by randomized pivot selection or median-of-three partitioning).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Worst case is $O(n^2)$, not $O(n \log n)$.
Option (B): Correct. Average is $O(n \log n)$ and worst is $O(n^2)$ on sorted/reverse-sorted inputs with fixed endpoint pivot.
Option (C): Incorrect. QuickSort worst case is quadratic, never cubic.
Option (D): Incorrect. Comparison sorts cannot achieve $O(n)$ average time.

HIGH-YIELD EXAM TAKEAWAY: QuickSort: Average $O(n \log n)$, Worst $O(n^2)$ on sorted arrays with naive pivot. Space: $O(\log n)$ stack.

Question 64 7.2 Graph Algorithms: Minimum Spanning Trees

Which design paradigm do both Prim's and Kruskal's Minimum Spanning Tree (MST) algorithms follow, and which auxiliary data structure allows Kruskal's algorithm to achieve an efficient $O(E \log E)$ time complexity?

✓ **Correct Answer: (C) Greedy Method; Disjoint Set Union (Union-Find) with path compression**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Minimum Spanning Tree algorithms:
1. Both Prim's and Kruskal's algorithms follow the **Greedy algorithmic paradigm**: at every step, they make a locally optimal choice (lowest weight edge) that leads to a globally optimal spanning tree.
2. **Kruskal's Algorithm**:
- Sorts all E edges by weight: $O(E \log E)$.
- Iterates through sorted edges and adds an edge if and only if it does not form a cycle with already selected edges.
- Cycle detection is performed using **Disjoint Set Union (DSU / Union-Find)** with path compression and union by rank. Find and Union operations take nearly constant $O(\alpha(V))$ amortized time.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. MST algorithms do not use dynamic programming.
Option (B): Incorrect. Divide and conquer is used in MergeSort, not Kruskal's.
Option (C): Correct. Both algorithms are Greedy; Kruskal uses DSU (Union-Find) for cycle detection.
Option (D): Incorrect. Backtracking tests all combinations, which is unnecessary here.

HIGH-YIELD EXAM TAKEAWAY: Kruskal's & Prim's are Greedy algorithms. Kruskal sorts edges and detects cycles using Disjoint Set Union ($O(E \log E)$).

Question 65 7.3 Database Normalization: Normal Forms

A relational database schema $R(A, B, C, D)$ is in **Third Normal Form (3NF) if for every non-trivial functional dependency $X \to Y$:**

✓ **Correct Answer: (B) Either X is a superkey, or Y is a prime attribute (a member of a candidate key)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Relational Database Normal Forms:
- **1NF**: Attribute values are atomic; no repeating groups.
- **2NF**: In 1NF and no non-prime attribute is partially dependent on any candidate key (full functional dependency).
- **3NF**: In 2NF and for every functional dependency $X \to Y$:
1. $X \to Y$ is trivial ($Y \subseteq X$), OR
2. **X is a superkey**, OR
3. **Y is a prime attribute** (part of some candidate key).
(Condition 3 eliminates transitive dependencies while preserving dependencies during decomposition).
- **BCNF (Boyce-Codd)**: Stricter than 3NF; requires that for every $X \to Y$, **X must be a superkey** (Option A).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Requiring X to be a superkey exclusively defines **BCNF**.
Option (B): Correct. 3NF allows X to be a superkey OR Y to be a prime attribute.
Option (C): Incorrect. Partial dependencies violate 2NF.
Option (D): Incorrect. Irrelevant to 3NF.

HIGH-YIELD EXAM TAKEAWAY: 3NF rule: In $X \to Y$, either X is a superkey OR Y is a prime attribute. BCNF rule: X MUST be a superkey.

Question 66 7.3 SQL & Query Processing

In an SQL query containing `SELECT`, `FROM`, `WHERE`, `GROUP BY`, and `HAVING` clauses, what is the exact logical execution order of the filtering and grouping clauses?

✓ **Correct Answer: (B) `WHERE` → `GROUP BY` → `HAVING`**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Logical SQL Query Execution Order:
1. ** `FROM` / `JOIN` **: Load tables and perform cartesian products/joins.
2. ** `WHERE` **: Filter individual rows before any grouping occurs (cannot use aggregate functions).
3. ** `GROUP BY` **: Group the remaining rows based on common column values.
4. ** `HAVING` **: Filter the aggregated groups (evaluates conditions on aggregate functions like `COUNT()`, `AVG()`).
5. ** `SELECT` **: Project the requested columns and evaluate expressions.
6. ** `DISTINCT` **: Remove duplicates.
7. ** `ORDER BY` **: Sort final result set.
8. ** `LIMIT` / `OFFSET` **: Restrict output row count.
Therefore, the filtering sequence is strictly: ** `WHERE` → `GROUP BY` → `HAVING` **.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Groups cannot be filtered before they are formed.
Option (B): Correct. Rows are filtered (`WHERE`), then grouped (`GROUP BY`), then groups are filtered (`HAVING`).
Option (C): Incorrect. `WHERE` filters rows before grouping, not after.
Option (D): Incorrect. `HAVING` operates on groups created by `GROUP BY`.

HIGH-YIELD EXAM TAKEAWAY: SQL execution pipeline: `FROM` → `WHERE` (row filter) → `GROUP BY` → `HAVING` (group filter) → `SELECT`.

Question 67 7.4 Concurrency Control: Two-Phase Locking (2PL)

What formal theoretical guarantee is provided by the **Two-Phase Locking (2PL) protocol in relational database transaction management?**

✓ Correct Answer: **(B) It guarantees conflict serializability of concurrent transaction execution schedules**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Two-Phase Locking (2PL) rules and guarantees:
- **Phase 1 (Growing Phase)**: A transaction may acquire locks, but cannot release any lock.
- **Phase 2 (Shrinking Phase)**: A transaction may release locks, but cannot acquire any new lock.
- **Theoretical Guarantee**: Any execution schedule produced by 2PL is guaranteed to be **Conflict Serializable** (the precedence serialization graph has no directed cycles).
- **Limitations**: Standard 2PL **does NOT prevent deadlocks** (transactions can still enter circular lock waits). Strict 2PL or Rigorous 2PL prevents cascading rollbacks by holding exclusive locks until transaction commit.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 2PL can still deadlock (e.g. T_1 holds lock on A and needs B; T_2 holds B and needs A).

Option (B): Correct. 2PL mathematically guarantees conflict serializability.

Option (C): Incorrect. 2PL is a concurrency control protocol, not a storage engine optimization.

Option (D): Incorrect. Transactions can be aborted to break deadlocks.

HIGH-YIELD EXAM TAKEAWAY: Two-Phase Locking (2PL) guarantees conflict serializability, but does NOT prevent deadlocks.

Question 68 7.5 Operating Systems: Process Scheduling

Three processes P_1, P_2, P_3 arrive simultaneously at time $t = 0$ ms with CPU burst times of 8 ms, 4 ms, and 2 ms, respectively. Using non-preemptive **Shortest Job First (SJF) scheduling, what is the average waiting time across all three processes?**

✓ Correct Answer: **(A) 2.67 ms**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

SJF scheduling schedules processes in ascending order of burst time:
1. Execution order: P_3 (burst 2) $\rightarrow$ P_2 (burst 4) $\rightarrow$ P_1 (burst 8).
2. Gantt Chart:
- $t = 0$ to $t = 2$: Process P_3 executes.
- $t = 2$ to $t = 6$: Process P_2 executes.
- $t = 6$ to $t = 14$: Process P_1 executes.
3. Individual Waiting Times ($\text{Waiting Time} = \text{Start Time} - \text{Arrival Time}$):
- P_3 waiting time $= 0 - 0 = 0$ ms
- P_2 waiting time $= 2 - 0 = 2$ ms
- P_1 waiting time $= 6 - 0 = 6$ ms
4. Average Waiting Time:

$$\text{AWT} = \frac{0 + 2 + 6}{3} = \frac{8}{3} \approx \mathbf{2.67}\ \text{ms}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Individual waiting times are 0, 2, and 6 ms. Average $= 8/3 \approx 2.67$ ms.

Option (B): Incorrect. Arithmetic error.

Option (C): Incorrect. Corresponds to average turnaround time ($[2 + 6 + 14]/3 = 7.33$ or partial waiting time).

Option (D): Incorrect. Corresponds to FCFS scheduling ($P_1 \rightarrow P_2 \rightarrow P_3$: AWT $= [0 + 8 + 12]/3 = 6.67$ ms).

HIGH-YIELD EXAM TAKEAWAY: SJF minimizes average waiting time by executing shortest burst jobs first.

Question 69 7.5 Operating Systems: Deadlock Conditions

Which of the following is NOT one of the four necessary and sufficient Coffman conditions required for an operating system deadlock to occur?

✓ Correct Answer: **(C) Preemption of resources by the operating system**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Coffman Conditions for Deadlock (all four must hold simultaneously):
1. **Mutual Exclusion**: At least one resource must be held in a non-shareable mode (only one process at a time).
2. **Hold and Wait**: A process must currently hold at least one resource and be waiting to acquire additional resources held by other processes.
3. **No Preemption**: Resources cannot be forcibly preempted from a process; they can only be released voluntarily by the holding process after completing its task.
4. **Circular Wait**: A closed chain of processes exists $\{P_0, P_1, \ldots, P_n\}$ such that P_0 is waiting for a resource held by P_1, P_1 waits for P_2, ..., and P_n waits for P_0.
*(Notice that 'Preemption' actively breaks deadlock; the condition is strictly **No Preemption**)*.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Mutual Exclusion is a mandatory Coffman condition.
Option (B): Incorrect. Hold and Wait is a mandatory Coffman condition.
Option (C): Correct. The condition is **No Preemption**. If the OS can preempt resources, deadlocks cannot persist.
Option (D): Incorrect. Circular Wait is a mandatory Coffman condition.

HIGH-YIELD EXAM TAKEAWAY: Deadlock requires 4 simultaneous conditions: Mutual Exclusion, Hold & Wait, **No Preemption**, and Circular Wait.

Question 70 7.6 Memory Management: Belady's Anomaly

Which page replacement algorithm is susceptible to **Belady's Anomaly (where increasing the number of allocated physical page frames paradoxically results in an increased number of page faults)?**

✓ Correct Answer: **(C) First-In, First-Out (FIFO)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Belady's Anomaly in Virtual Memory:
- Intuitively, allocating more physical memory frames should decrease (or at least not increase) the number of page faults.
- In 1969, László Bélády proved that under the **FIFO (First-In, First-Out)** page replacement algorithm, certain reference strings (e.g. `1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5`) produce **more** page faults with 4 frames (10 faults) than with 3 frames (9 faults).
- **Stack Algorithms**: Algorithms where the set of pages in memory for m frames is always a strict subset of the pages for $m+1$ frames (such as **LRU** and **Optimal**) can mathematically **never** experience Belady's anomaly.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. LRU is a stack algorithm and is mathematically immune to Belady's anomaly.
Option (B): Incorrect. Optimal is a stack algorithm and never suffers from Belady's anomaly.
Option (C): Correct. FIFO is not a stack algorithm and exhibits Belady's anomaly.
Option (D): Incorrect. Stack-like aging algorithms prevent the anomaly.

HIGH-YIELD EXAM TAKEAWAY: FIFO page replacement suffers from Belady's Anomaly. Stack algorithms (LRU, Optimal) never experience it.

SECTION VIII: SOFTWARE ENGINEERING AND OOAD

Solutions Q.71 – Q.80

Question 71

8.1 Software Process Models: The Spiral Model

What is the primary distinguishing feature and fundamental advantage of Barry Boehm's **Spiral Model of software development compared to the classic Waterfall model?**

✓ Correct Answer: **(B) Explicit, systematic risk analysis and mitigation performed at each iterative cycle**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The Spiral Model (Barry Boehm, 1986):
- It is an evolutionary, risk-driven software development process model that combines the iterative nature of prototyping with the controlled aspects of the waterfall model.
- The model is visualized as a spiral with four repeating quadrants in each loop:
1. Determine objectives, alternatives, and constraints.
2. **Identify and resolve risks** (technical, financial, operational) through prototyping and simulation.
3. Develop and verify the next-level product.
4. Plan the next phase.
- Its primary advantage is systematic **risk management** before committing major resources.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Describes rigid misuse of the waterfall model.
Option (B): Correct. Systematic risk analysis at each cycle is the defining hallmark of the Spiral Model.
Option (C): Incorrect. The spiral model incorporates rigorous engineering documentation.
Option (D): Incorrect. The spiral model is primarily intended for large, high-risk, mission-critical systems.

HIGH-YIELD EXAM TAKEAWAY: Spiral Model = Evolutionary + Risk-Driven. Every cycle explicitly conducts risk analysis and prototyping.

Question 72

8.1 Requirements Engineering: Functional vs Non-Functional

Which of the following requirements in a Software Requirements Specification (SRS) is classified as a **Non-Functional Requirement (NFR)?**

✓ Correct Answer: **(C) "The system shall process and return 95% of database search queries within 200 milliseconds under a concurrent load of 1,000 active users."**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Functional vs Non-Functional Requirements:
- **Functional Requirements (FR)**: Specify **what** the system must do—the functional features, behaviors, user interactions, services, and calculations (Options A, B, and D represent functional requirements).
- **Non-Functional Requirements (NFR)**: Specify **how well** the system must perform—the operational constraints, quality attributes, performance thresholds, response time, scalability, security standards, reliability, and availability.
Option C specifies response latency and concurrency limits, making it a classic **performance non-functional requirement**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. User registration is a functional feature.
Option (B): Incorrect. Report generation is a functional feature.
Option (C): Correct. Query latency under concurrent load is a performance non-functional requirement.
Option (D): Incorrect. Role management is a functional capability.

HIGH-YIELD EXAM TAKEAWAY: Functional = features and behavior (what it does). Non-Functional = quality and constraints (speed, security, uptime).

Question 73 8.2 Software Design: Coupling and Cohesion

In modular software architectural design, high-quality, maintainable, and reusable systems should strive for:

✓ **Correct Answer: (D) Low coupling and high cohesion**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Coupling and Cohesion principles:
- **Cohesion**: Measures the degree of functional relatedness and focus of responsibilities within a single module. High cohesion (especially **functional cohesion**) means a module does one thing and does it well, making it easy to test, debug, and understand.
- **Coupling**: Measures the degree of interdependence and connectivity between different modules. Low coupling (**loose coupling**, e.g., data coupling via parameters) minimizes ripple effects: modifying one module does not break other modules.
- Golden rule of software engineering: **High Cohesion and Low Coupling**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. High coupling makes modules brittle and difficult to change.
Option (B): Incorrect. Low cohesion means modules lack clear focus and contain scattered logic.
Option (C): Incorrect. This represents the worst possible architectural antipattern.
Option (D): Correct. High cohesion (focused modules) and low coupling (minimal inter-dependencies) is the gold standard.

HIGH-YIELD EXAM TAKEAWAY: Good software architecture maximizes cohesion (internal unity) and minimizes coupling (external interdependence).

Question 74 8.3 Software Testing: Cyclomatic Complexity

For a software program whose control flow graph G contains $E = 14$ edges, $N = 10$ nodes, and $P = 1$ connected component, what is McCabe's Cyclomatic Complexity $V(G)$, representing the number of linearly independent execution paths?

✓ **Correct Answer: (B) 6**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

McCabe's Cyclomatic Complexity formulas:
1. Using edges and nodes:

$$V(G) = E - N + 2P$$

Where $E = 14$ edges, $N = 10$ nodes, and $P = 1$ component:

$$V(G) = 14 - 10 + 2(1) = 4 + 2 = \mathbf{6}$$

2. Alternatively, using predicate (decision) nodes D:

$$V(G) = D + 1$$

Here $D = 5$ conditional decisions, giving $5 + 1 = 6$.
This means there are exactly 6 linearly independent execution paths through the code, defining the minimum number of test cases required for complete basis path coverage.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $14 - 10 = 4$ forgets to add $+2P$.
Option (B): Correct. $V(G) = 14 - 10 + 2(1) = 6$.
Option (C): Incorrect. 14 is the number of edges.
Option (D): Incorrect. Summing edges and nodes.

HIGH-YIELD EXAM TAKEAWAY: Cyclomatic Complexity: $V(G) = E - N + 2P = \text{Predicate Nodes } D + 1 = \text{Number of independent paths.}$

Question 75 8.3 Software Testing: Boundary Value Analysis

If an online admission form input field accepts student age as an integer between 18 and 60 (inclusive), which set of test input values represents rigorous **Boundary Value Analysis (BVA)?**

✓ **Correct Answer: (B) {17, 18, 19, 59, 60, 61}**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Boundary Value Analysis (BVA):
- Software errors occur disproportionately at the boundaries of input domains (off-by-one errors, `<` vs `<=`).
- For a valid input range $[A, B]$ ($18 \leq \text{age} \leq 60$):
- Lower boundary testing: Just below boundary ($A - 1 = 17$), exact boundary ($A = 18$), and just above boundary ($A + 1 = 19$).
- Upper boundary testing: Just below boundary ($B - 1 = 59$), exact boundary ($B = 60$), and just above boundary ($B + 1 = 61$).
- The complete set of 6 boundary test values is **$\{17, 18, 19, 59, 60, 61\}$**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. These are interior values used in Equivalence Partitioning, not boundary values.
Option (B): Correct. Tests boundary edges: $18 - 1, 18, 18 + 1$ and $60 - 1, 60, 60 + 1$.
Option (C): Incorrect. Values are far from the valid boundaries.
Option (D): Incorrect. Missing boundary neighbors.

HIGH-YIELD EXAM TAKEAWAY: Boundary Value Analysis tests: $\{\text{Min} - 1, \text{Min}, \text{Min} + 1\}$ and $\{\text{Max} - 1, \text{Max}, \text{Max} + 1\}$.

Question 76 8.3 Software Cost Estimation: COCOMO Model

In the Constructive Cost Model (COCOMO), a software project undertaken by a small, experienced in-house team with flexible requirements in a familiar hardware environment is categorized under which project mode?

✓ **Correct Answer: (C) Organic mode**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

COCOMO Project Modes (Boehm):
1. **Organic Mode**: Relatively small, simple software projects (typically < 50 KLOC) developed by experienced teams in a familiar in-house environment with relaxed, flexible requirements. Low complexity.
2. **Semidetached Mode**: Intermediate projects ($50 - 300$ KLOC) where team members possess mixed experience and requirements are moderately rigid.
3. **Embedded Mode**: Complex software developed under tight hardware constraints, rigid real-time requirements, and strict regulations where changing interfaces is costly (e.g., flight control software, medical devices).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Embedded mode operates under tight constraints and rigid requirements.
Option (B): Incorrect. Semidetached mode represents intermediate complexity.
Option (C): Correct. Small team, familiar environment, and flexible requirements define Organic mode.
Option (D): Incorrect. Not a COCOMO mode.

HIGH-YIELD EXAM TAKEAWAY: COCOMO Modes: Organic (small, familiar, flexible), Semidetached (intermediate), Embedded (tight constraints, complex).

Question 77 8.3 Quality Management: CMMI Levels

In the Capability Maturity Model Integration (CMMI), at which maturity level are software engineering and management processes standardized, documented, and established across the entire organization?

✓ Correct Answer: **(C) Level 3 (Defined)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

CMMI 5 Maturity Levels:
1. **Level 1 (Initial)**: Processes are unpredictable, ad-hoc, chaotic; success depends on individual heroics.
2. **Level 2 (Managed)**: Processes are planned and executed at the **individual project level**; basic project management disciplines are maintained.
3. **Level 3 (Defined)**: Processes are standardized, well-documented, and institutionalized across the **entire organization**; standard organizational processes are tailored for specific projects.
4. **Level 4 (Quantitatively Managed)**: Processes are controlled using statistical and quantitative measurement techniques.
5. **Level 5 (Optimizing)**: Focus on continuous process improvement through innovative technological enhancements.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Level 1 is ad-hoc and chaotic.
Option (B): Incorrect. Level 2 achieves discipline at the project level, not organization-wide.
Option (C): Correct. Level 3 (Defined) establishes standard processes across the entire organization.
Option (D): Incorrect. Level 5 focuses on continuous quantitative optimization.

HIGH-YIELD EXAM TAKEAWAY: CMMI Levels: 1-Initial, 2-Managed (project level), 3-Defined (organization-wide), 4-Quantitatively Managed, 5-Optimizing.

Question 78 8.4 OOAD & UML: Class Relationships

In a UML Class Diagram, what is the semantic difference between **Composition (filled black diamond) and **Aggregation** (hollow/white diamond)?**

✓ Correct Answer: **(B) Composition represents strong ownership where the lifetime of the part is tied to the whole (if the whole is destroyed, parts are destroyed); Aggregation represents a weaker 'has-a' relationship where parts can exist independently**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

UML Association Types:
- **Composition (Filled Black Diamond)**: Strong 'part-of' relationship with **coincident lifetimes**. The child object cannot exist without the parent. For example: A `House` is composed of `Rooms`. If the `House` is demolished, the `Rooms` cease to exist. A part can belong to only one whole at a time.
- **Aggregation (Hollow/White Diamond)**: Weak 'has-a' relationship where the child object can exist independently of the parent. For example: A `Department` aggregates `Professors`. If the `Department` is closed, the `Professors` continue to exist and can join other departments.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Reverses the definitions.
Option (B): Correct. Composition = strong ownership and shared lifetime; Aggregation = weak relationship with independent lifetimes.
Option (C): Incorrect. Neither represents inheritance (generalization).
Option (D): Incorrect. The distinction is a core UML design concept.

HIGH-YIELD EXAM TAKEAWAY: Composition (solid diamond): part dies with whole. Aggregation (open diamond): part survives deletion of whole.

Question 79 8.4 OOAD & UML: Use Case Diagrams

In UML Use Case diagrams, what is the key distinction between the `<<include>>` and `<<extend>>` relationships between two use cases?

✓ Correct Answer: **(B) `<<include>>` indicates that the base use case unconditionally incorporates the behavior of the included use case; `<<extend>>` indicates that an extending use case conditionally adds behavior at a defined extension point**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

UML Use Case Relationships:
- ** `<<include>>` **: Represents mandatory, unconditional behavior shared among multiple use cases. Whenever the base use case executes, the included use case is **always executed** (e.g., `Withdraw Cash` <<include>> `Authenticate User`). Arrow points from base to included use case.
- ** `<<extend>>` **: Represents optional, conditional behavior that executes only when specific triggering criteria are met at an extension point (e.g., `Print Receipt` <<extend>> `Withdraw Cash` only if paper is available and user requests it). Arrow points from extending use case to base use case.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Reverses the mandatory vs optional nature of the relationships.
Option (B): Correct. `<<include>>` is mandatory and unconditional; `<<extend>>` is optional and conditional.
Option (C): Incorrect. Actors connect via Association lines, not include/extend stereotypes.
Option (D): Incorrect. They have distinct semantics.

HIGH-YIELD EXAM TAKEAWAY: `<<include>>` = mandatory common behavior (always runs). `<<extend>>` = optional conditional behavior (runs only on condition).

Question 80 8.3 Software Configuration Management (SCM)

In Software Configuration Management (SCM), a **Baseline is formally defined as:**

✓ Correct Answer: **(B) A formally reviewed and agreed-upon version of a software configuration item that serves as the basis for further development and can only be modified through formal change control procedures**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Software Configuration Management Baseline:
- An SCM **Baseline** (e.g., Requirements Baseline, Design Baseline, Release Baseline) is a software work product (specification, code, documentation) that has undergone formal technical review and approval.
- Once established, it is frozen and put under formal configuration control.
- No changes can be made to a baselined item without submitting a formal **Change Request (CR)**, which must be evaluated, approved, and tracked by the **Change Control Board (CCB)**.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Brainstorming notes are informal and unreviewed.
Option (B): Correct. A baseline is an approved reference point changeable only through formal change management.
Option (C): Incorrect. Local developer copies are uncontrolled working versions.
Option (D): Incorrect. Hardware requirements define system specifications, not SCM baselines.

HIGH-YIELD EXAM TAKEAWAY: Baseline = formally approved software artifact that serves as a foundation and requires formal CCB approval to modify.

SECTION IX: ARTIFICIAL INTELLIGENCE AND NEURAL NETWORKS

Solutions Q.81 – Q.90

Question 81

9.1 AI Foundations & Intelligent Agents

In Artificial Intelligence, an environment that is fully observable, deterministic, sequential, static, discrete, and single-agent is best exemplified by which of the following task domains?

✓ Correct Answer: **(C) Solving an 8-Puzzle sliding tile problem**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Analyzing the 8-Puzzle environment characteristics:
1. **Fully Observable**: The agent's sensors view the complete board state at all times.
2. **Deterministic**: Every tile move deterministically yields a known next state.
3. **Sequential**: Current tile moves affect all subsequent available moves.
4. **Static**: The board does not change while the agent is deliberating.
5. **Discrete**: There are a finite number of distinct board states and moves.
6. **Single-Agent**: Only one entity manipulates the board (unlike Chess, which is multi-agent competitive).
Autonomous driving is partially observable, stochastic, continuous, and dynamic.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Driving is partially observable, stochastic, dynamic, continuous, and multi-agent.
Option (B): Incorrect. Chess is a multi-agent environment, and speed chess is semi-dynamic due to the clock.
Option (C): Correct. 8-Puzzle is fully observable, deterministic, sequential, static, discrete, and single-agent.
Option (D): Incorrect. Medical diagnosis is partially observable, stochastic, and dynamic.

HIGH-YIELD EXAM TAKEAWAY: 8-Puzzle = Fully Observable, Deterministic, Sequential, Static, Discrete, Single-Agent.

Question 82

9.2 Uninformed Search: Iterative Deepening Search

Why is **Iterative Deepening Search (IDS) widely regarded as the preferred uninformed search strategy for large state spaces with unknown solution depth?**

✓ Correct Answer: **(A) It combines the completeness and optimality of Breadth-First Search with the low linear memory complexity $O(bd)$ of Depth-First Search**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Iterative Deepening Search (IDS / IDDFS):
- **Breadth-First Search (BFS)** is complete and optimal (for unit step costs), but requires **$O(b^d)$ exponential memory**, exhausting physical RAM even at modest depths.
- **Depth-First Search (DFS)** requires only **$O(bd)$ linear memory**, but is not complete (can get trapped in infinite paths) and is not optimal.
- **IDS** repeatedly executes Depth-Limited DFS, increasing the depth limit $(0, 1, 2, \ldots, d)$ in each iteration:
- Completeness: **Yes** (guaranteed to find a solution if one exists).
- Optimality: **Yes** (finds the shallowest goal first, like BFS).
- Space Complexity: **$O(bd)$ linear memory** (like DFS).
- Overhead: Top levels are repeated, but because the tree grows exponentially (b^d), the last level dominates ($> 85\%$ of nodes), keeping time complexity at $O(b^d)$.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. IDS combines BFS's completeness/optimality with DFS's modest $O(bd)$ memory usage.
Option (B): Incorrect. Caching all nodes would cause exponential memory consumption.
Option (C): Incorrect. IDS is an uninformed (blind) search; it uses no heuristics.
Option (D): Incorrect. Bidirectional search searches backwards, not IDS.

HIGH-YIELD EXAM TAKEAWAY: Iterative Deepening Search (IDS) achieves BFS optimality/completeness with DFS linear space $O(bd)$.

Question 83 9.2 Informed Search: A* Algorithm & Heuristics

In A* search, an evaluation function is defined as $f(n) = g(n) + h(n)$. To guarantee that A* tree search finds an optimal (shortest-cost) path to the goal, the heuristic function $h(n)$ must be **admissible, meaning:**

✓ **Correct Answer: (B) $h(n)$ must never overestimate the true minimal cost to reach the goal ($0 \leq h(n) \leq h^*(n)$)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

A* Heuristic Properties:
1. The evaluation function is:

$$f(n) = g(n) + h(n)$$

where $g(n)$ is the exact cost from the start node to node n, and $h(n)$ is the estimated cost from node n to the goal.
2. **Admissibility**: A heuristic $h(n)$ is admissible if it is optimistic—it **never overestimates** the true minimal cost $h^*(n)$ to reach the goal:

$$0 \leq h(n) \leq h^*(n) \quad \forall n$$

If $h(n)$ never overestimates, A* tree search is mathematically guaranteed to return the optimal path.
*(For graph search with closed lists, the heuristic must additionally be **consistent / monotonic**: $h(n) \leq c(n, a, n') + h(n')$)*.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Underestimating is allowed and required; overestimating violates admissibility.
Option (B): Correct. Admissibility requires $h(n) \leq h^*(n)$ (optimistic, never overestimates).
Option (C): Incorrect. $g(n)$ is past cost, while $h(n)$ estimates future cost.
Option (D): Incorrect. At the goal node, $h(\text{goal}) = 0$.

HIGH-YIELD EXAM TAKEAWAY: A* tree search is optimal if $h(n)$ is admissible ($h(n) \leq h^*(n)$, never overestimates true cost).

Question 84 9.2 Adversarial Search: Alpha-Beta Pruning

In the Minimax game-playing search algorithm with Alpha-Beta pruning, an **Alpha Cut-off occurs at a MIN node when:**

✓ **Correct Answer: (A) $\beta \leq \alpha$, where α is the best (highest) value found so far by MAX, and β is the best (lowest) value found so far by MIN**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Alpha-Beta Pruning principles:
- **α**: The value of the best (highest-value) choice found so far along the path for MAX (initialized to $-\infty$).
- **β**: The value of the best (lowest-value) choice found so far along the path for MIN (initialized to $+\infty$).
- **Pruning condition**: When searching a MIN node, as soon as its evaluated value becomes less than or equal to α ($\beta \leq \alpha$):
- MAX already has a guaranteed option higher up in the game tree with value α.
- MAX will never choose the branch leading to this MIN node because MIN can force a score $\leq \alpha$.
- Therefore, the remaining child branches of this MIN node can be **pruned** without affecting the final minimax decision.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. When $\beta \leq \alpha$ at a MIN node, an alpha cut-off occurs.

Option (B): Incorrect. Pruning depends on relative comparison $\beta \leq \alpha$, not zero values.

Option (C): Incorrect. Does not cause pruning.

Option (D): Incorrect. Alpha-beta pruning applies at any depth.

HIGH-YIELD EXAM TAKEAWAY: Alpha-Beta Pruning: Prune remaining siblings whenever $\beta \leq \alpha$ (alpha cut-off at MIN, beta cut-off at MAX).

Question 85 9.3 Knowledge Representation: First-Order Logic & Unification

In First-Order Predicate Logic (FOPL) automated theorem proving, what is the Most General Unifier (MGU) that unifies the two atomic literals $P(x, f(y))$ and $P(a, f(g(z)))$?

✓ **Correct Answer: (B) $\{x/a, y/g(z)\}$**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Unification algorithm step by step:
We wish to find substitution θ such that $P(x, f(y))\theta = P(a, f(g(z)))\theta$:
1. Predicate symbols match: $P = P$.
2. Compare first arguments: x and a. Variable x unifies with constant a via substitution: $\{x/a\}$.
3. Compare second arguments: $f(y)$ and $f(g(z))$.
- Top-level function symbols match: $f = f$.
- Compare inner arguments: y and $g(z)$. Variable y unifies with term $g(z)$ via substitution: $\{y/g(z)\}$.
4. Combining substitutions yields the Most General Unifier (MGU):

$$\theta = \{\mathbf{x/a, y/g(z)}\}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Substituting y/z yields $f(z) \neq f(g(z))$.

Option (B): Correct. MGU is $\{x/a, y/g(z)\}$.

Option (C): Incorrect. Replaces x with $f(a)$ rather than a.

Option (D): Incorrect. Functions with matching names and arity unify if their arguments unify.

HIGH-YIELD EXAM TAKEAWAY: Unification finds variable substitutions to make two logic expressions identical: $P(x, f(y)) \equiv P(a, f(g(z))) \implies \{x/a, y/g(z)\}$.

Question 86 9.3 Probabilistic Reasoning: Bayes' Rule

A network diagnostic AI monitors server hardware failures. The prior probability of a rare hardware fault is $P(\text{Fault}) = 0.01$. An automated diagnostic test has a True Positive rate of $P(\text{Test}^+ \mid \text{Fault}) = 0.90$ and a False Positive rate of $P(\text{Test}^+ \mid \neg\text{Fault}) = 0.05$. If a server tests positive (Test^+), what is the posterior probability $P(\text{Fault} \mid \text{Test}^+)$ that the server actually has the fault?

✓ Correct Answer: **(C) 15.4%**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Application of Bayes' Theorem:

1. Given probabilities:
- $P(\text{Fault}) = 0.01 \implies P(\neg\text{Fault}) = 0.99$
- $P(\text{Test}^+ \mid \text{Fault}) = 0.90$
- $P(\text{Test}^+ \mid \neg\text{Fault}) = 0.05$

2. Calculate total probability of a positive test using the Law of Total Probability:

$$P(\text{Test}^+) = P(\text{Test}^+ \mid \text{Fault})P(\text{Fault}) + P(\text{Test}^+ \mid \neg\text{Fault})P(\neg\text{Fault})$$

$$P(\text{Test}^+) = (0.90 \times 0.01) + (0.05 \times 0.99) = 0.0090 + 0.0495 = 0.0585$$

3. Calculate posterior probability by Bayes' Rule:

$$P(\text{Fault} \mid \text{Test}^+) = \frac{P(\text{Test}^+ \mid \text{Fault})P(\text{Fault})}{P(\text{Test}^+)} = \frac{0.0090}{0.0585} = \frac{90}{585} \approx \mathbf{0.1538} = \mathbf{15.4\%}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Confuses the test sensitivity $P(\text{Test}^+ \mid \text{Fault}) = 90\%$ with the posterior probability.

Option (B): Incorrect. Ignores the base rate of the disease/fault.

Option (C): Correct. $P(\text{Fault} \mid \text{Test}^+) = \frac{0.009}{0.0585} \approx 15.4\%$.

Option (D): Incorrect. Calculation error.

HIGH-YIELD EXAM TAKEAWAY: Bayes' Rule: $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$. Due to low base rates (1%), most positive tests are false positives.

Question 87 9.5 Machine Learning: Decision Trees (ID3)

In inductive decision tree learning algorithms (such as ID3), the attribute chosen to split the dataset at each decision node is the one that achieves:

✓ **Correct Answer: (B) The highest Information Gain (maximum reduction in Entropy)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Decision Tree Splitting Criteria (ID3 algorithm):

1. Entropy measures the impurity or disorder of a dataset S:

$$H(S) = -\sum_{i=1}^{c} p_i \log_2(p_i)$$

2. **Information Gain** $\text{Gain}(S, A)$ measures the expected reduction in entropy achieved by partitioning the dataset using attribute A:

$$\text{Gain}(S, A) = H(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} H(S_v)$$

3. At each step, ID3 greedily selects the candidate attribute that **maximizes Information Gain**, creating subsets that are as homogenous (pure) as possible.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Minimum Information Gain preserves high impurity.

Option (B): Correct. ID3 selects the attribute that maximizes Information Gain (greatest entropy reduction).

Option (C): Incorrect. Selecting high-branching attributes arbitrarily leads to severe overfitting (fixed by Gain Ratio in C4.5).

Option (D): Incorrect. ID3 operates primarily on categorical attributes.

HIGH-YIELD EXAM TAKEAWAY: ID3 selects the feature that maximizes Information Gain (greatest reduction in Shannon Entropy).

Question 88 9.5 Machine Learning: Naive Bayes Classifier

Why is the Naive Bayes classification algorithm termed 'naive'?

✓ **Correct Answer: (B) It makes the simplifying assumption that all feature attributes are conditionally independent of each other given the class label**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

The 'Naive' Assumption in Naive Bayes:

- In Bayesian classification, calculating the true joint probability $P(x_1, x_2, \ldots, x_d \mid C)$ requires learning an intractable number of conditional dependencies.
- The Naive Bayes classifier makes the strong ('naive') assumption that **all feature attributes are conditionally independent given the class label C**:

$$P(x_1, x_2, \ldots, x_d \mid C) = \prod_{i=1}^{d} P(x_i \mid C)$$

- Although this independence assumption is rarely true in real-world data (features often correlate), Naive Bayes performs remarkably well in practice (e.g. text classification, spam filtering) and scales linearly ($O(d)$).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Gaussian Naive Bayes is just one continuous variant.
Option (B): Correct. It is called naive because it assumes all features are conditionally independent given the class.
Option (C): Incorrect. Naive Bayes natively supports multi-class classification.
Option (D): Incorrect. It computes prior and conditional probabilities from training data.

HIGH-YIELD EXAM TAKEAWAY: Naive Bayes is 'naive' because it assumes complete conditional independence among all features given the class.

Question 89 9.6 Neural Networks: The Perceptron

In their landmark 1969 book, Marvin Minsky and Seymour Papert mathematically proved that a single-layer perceptron with a threshold activation function cannot solve which classic logical problem?

✓ **Correct Answer: (D) Logical XOR (Exclusive-OR)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Linear Separability and the Perceptron:
- A single-layer artificial perceptron calculates a weighted sum: $z = w_1x_1 + w_2x_2 + b$.
- The decision boundary generated is a linear hyperplane ($w_1x_1 + w_2x_2 + b = 0$).
- A single-layer perceptron can only classify **linearly separable** functions (AND, OR, NOT are all linearly separable).
- For **XOR**, the outputs are:
- $(0,0) \to 0$, $(1,1) \to 0$
- $(0,1) \to 1$, $(1,0) \to 1$
- No single straight line can separate the `1` outputs from the `0` outputs in 2D space. XOR is **not linearly separable**. Solving XOR requires a **Multi-Layer Perceptron (MLP)** with at least one hidden layer and non-linear activation functions.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. AND is linearly separable.
Option (B): Incorrect. OR is linearly separable.
Option (C): Incorrect. NOT is a 1D linearly separable inverter.
Option (D): Correct. Minsky & Papert proved that a single-layer perceptron cannot compute XOR because it is not linearly separable.

HIGH-YIELD EXAM TAKEAWAY: Single-layer perceptron can only solve linearly separable problems. It cannot solve XOR; MLPs are required.

Question 90 9.6 Neural Networks: Backpropagation Algorithm

When training a Multi-Layer Perceptron (MLP) using the Backpropagation algorithm, why must the activation functions used in hidden layer neurons be continuous and differentiable?

✓ Correct Answer: **(A) Because the algorithm uses the mathematical chain rule of calculus to compute partial derivatives (gradients) of the loss function with respect to network weights**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Why differentiability is mandatory in Backpropagation:
1. The Backpropagation algorithm optimizes weights using **Gradient Descent**:

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}}$$

2. To propagate the error gradient backward from the output layer through hidden layers, the calculus **chain rule** is applied:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial net_j} \times \frac{\partial net_j}{\partial w_{ij}} = \delta_j \cdot x_i$$

where δ_j involves the first derivative of the activation function: $f'(net_j)$.
3. If an activation function is non-differentiable (like the Heaviside step function, whose derivative is 0 everywhere and undefined at 0), gradient descent cannot proceed. Differentiable functions like **Sigmoid**, **Tanh**, and **ReLU** allow gradient computation.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Gradient descent and the chain rule require evaluating activation function derivatives at each layer.
Option (B): Incorrect. Step functions are computationally cheap, but have zero derivative.
Option (C): Incorrect. Learning rate is a user hyperparameter.
Option (D): Incorrect. Weights are not constrained to sum to zero.

HIGH-YIELD EXAM TAKEAWAY: Backpropagation requires differentiable activation functions to evaluate error gradients via the chain rule.

SECTION X: PROJECT PLANNING, DESIGN AND IMPLEMENTATION

Solutions Q.91 – Q.100

Question 91

10.1 Engineering Drawings: Standard Sheet Sizes

According to ISO 216 and standard engineering drafting specifications, what is the exact surface area and trimmed dimensions of an **A0 drawing sheet, and what constant aspect ratio is preserved across all A-series sheets?**

✓ **Correct Answer: (B) Area = 1.00 m², Dimensions: 841 × 1189 mm, Aspect Ratio: 1:√2**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

ISO 216 Drafting Standards for A-Series Sheets:

1. The baseline sheet **A0** is defined to have a surface area of exactly **$1.00\ \text{m}^2$** ($10^6\ \text{mm}^2$).
2. The aspect ratio between width (x) and length (y) is strictly **$1 : \sqrt{2}$** ($x : y = 1 : \sqrt{2} \implies y = x\sqrt{2}$).
3. Solving for dimensions:

$$x \times (x\sqrt{2}) = 10^6 \implies x^2 = \frac{10^6}{\sqrt{2}} = 707106 \implies x \approx 841\ \text{mm}$$

$$y = 841 \times 1.4142 \approx 1189\ \text{mm}$$

4. Halving an A0 sheet along its length yields an A1 sheet ($594 \times 841\ \text{mm}$), halving A1 gives A2, halving A2 gives A3, and halving A3 gives standard A4 ($210 \times 297\ \text{mm}$), all preserving the constant $1 : \sqrt{2}$ aspect ratio.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $594 \times 841\ \text{mm}$ with $0.5\ \text{m}^2$ area is the **A1** sheet.

Option (B): Correct. A0 area is $1.00\ \text{m}^2$, dimensions are $841 \times 1189\ \text{mm}$, and aspect ratio is $1 : \sqrt{2}$.

Option (C): Incorrect. Arbitrary dimensions.

Option (D): Incorrect. Aspect ratio is $1 : \sqrt{2}$, not $1 : \sqrt{3}$.

HIGH-YIELD EXAM TAKEAWAY: ISO A0 sheet: Area $= 1.00\ \text{m}^2$, Dimensions $= 841 \times 1189\ \text{mm}$, Aspect Ratio $= 1 : \sqrt{2}$.

Question 92 10.2 Engineering Economics: Time Value of Money

An IT engineering enterprise invests NPR 1,000,000 today in enterprise cloud infrastructure. If the annual market interest rate is 10% compounded annually, what is the future worth (F) of this capital investment at the end of 3 years?

✓ **Correct Answer: (B) NPR 1,331,000**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Compound Interest Formula in Engineering Economics:

$$F = P(1+i)^n$$

Where:
- Present Worth $P = \text{NPR } 1,000,000$
- Interest rate $i = 10\% = 0.10$
- Number of compounding periods $n = 3 \text{ years}$

Calculating:

$$F = 1,000,000 \times (1+0.10)^3 = 1,000,000 \times (1.10)^3$$

$$(1.10)^3 = 1.331$$

$$F = 1,000,000 \times 1.331 = \text{NPR } \mathbf{1,331,000}$$

(Note: Simple interest would yield $1,000,000 \times [1+3(0.10)] = 1,300,000$, which ignores compound interest).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. NPR 1,300,000 is simple interest, not compound interest.
Option (B): Correct. $1,000,000 \times (1.1)^3 = \text{NPR } 1,331,000$.
Option (C): Incorrect. NPR 1,210,000 is the value after 2 years.
Option (D): Incorrect. Arbitrary estimate.

HIGH-YIELD EXAM TAKEAWAY: Future Worth formula: $F = P(1+i)^n$. Compound interest factor $(F/P, 10\%, 3) = 1.331$.

Question 93 10.2 Engineering Economics: Project Evaluation

In engineering economic feasibility appraisal, an independent commercial project is financially acceptable and viable if:

✓ Correct Answer: **(B) Net Present Value (NPV) > 0 and Internal Rate of Return (IRR) is greater than the Minimum Attractive Rate of Return (MARR)**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Project Evaluation Criteria in Engineering Economics:
1. **Net Present Value (NPV)**: $\text{NPV} = \sum_{t=0}^{n} \frac{CF_t}{(1+MARR)^t}$. A project is financially viable if **$\text{NPV} > 0$** (present worth of revenues exceeds present worth of costs).
2. **Internal Rate of Return (IRR)**: The break-even discount rate at which $\text{NPV} = 0$. A project is acceptable if **$\text{IRR} > \text{MARR}$** (the project's rate of return exceeds the company's cost of capital / hurdle rate).
3. **Benefit-Cost Ratio (B/C)**: Must satisfy **$B/C > 1.0$**.
4. **Payback Period**: Must be **less** than the project's useful economic life.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. $\text{NPV} < 0$ indicates financial loss.

Option (B): Correct. Financial viability requires $\text{NPV} > 0$ and $\text{IRR} > \text{MARR}$.

Option (C): Incorrect. Payback period must be shorter than project life.

Option (D): Incorrect. Operating profit must exceed depreciation for commercial viability.

HIGH-YIELD EXAM TAKEAWAY: Independent project acceptance criteria: $\text{NPV} > 0$, $\text{IRR} > \text{MARR}$, and $B/C > 1.0$.

Question 94 10.2 Depreciation Methods: Straight Line

A data center server cluster is procured for NPR 600,000. It has an estimated useful service life of 5 years and a salvage (scrap) value of NPR 100,000. Using the **Straight-Line Depreciation method, what is the annual depreciation charge and the book value of the cluster at the end of Year 2?**

✓ **Correct Answer: (A) Annual Depreciation: NPR 100,000; Book Value: NPR 400,000**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Straight-Line Depreciation calculation:
1. **Initial Cost (C)** $= \text{NPR } 600,000$.
2. **Salvage Value (S)** $= \text{NPR } 100,000$.
3. **Useful Life (N)** $= 5 \text{ years}$.
4. **Annual Depreciation Charge (D)**:

$$D = \frac{C - S}{N} = \frac{600,000 - 100,000}{5} = \frac{500,000}{5} = \text{NPR } \mathbf{100,000} \text{ per year}$$

5. **Accumulated Depreciation at end of Year 2 (D_2)**:

$$D_2 = 2 \times 100,000 = \text{NPR } 200,000$$

6. **Book Value at end of Year 2 (BV_2)**:

$$BV_2 = C - D_2 = 600,000 - 200,000 = \text{NPR } \mathbf{400,000}$$

.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Annual depreciation is NPR 100,000 and Book Value at Year 2 is NPR 400,000.
Option (B): Incorrect. Divides total cost by 5 without subtracting salvage value ($600,000/5 = 120,000$).
Option (C): Incorrect. Arithmetic calculation error.
Option (D): Incorrect. Subtracts only 1 year of depreciation instead of 2.

HIGH-YIELD EXAM TAKEAWAY: Straight-line depreciation: $D = \frac{C-S}{N}$. Book Value: $BV_t = C - (t \times D)$.

Question 95 10.3 Project Planning & Scheduling: CPM

In Critical Path Method (CPM) project network scheduling, what does the **Total Float (TF) of an activity represent?**

✓ Correct Answer: **(B) The maximum amount of time an activity can be delayed beyond its earliest start time without delaying the overall project completion date**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Float definitions in CPM Network Analysis:
1. **Total Float (TF)**:

$$TF = LS - ES = LF - EF$$

It represents the maximum amount of time an activity can be delayed beyond its Earliest Start (ES) without delaying the scheduled project completion date ($LF_{project}$). All activities lying on the **Critical Path** have a Total Float of zero ($TF = 0$).
2. **Free Float (FF)**:

$$FF = ES_{successor} - EF$$

It represents the time an activity can be delayed without delaying the earliest start of any immediate successor activity (Option A).
3. **Independent Float (IF)**: Delay allowed without affecting predecessors or successors.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. This is the formal definition of **Free Float (FF)**.
Option (B): Correct. Total Float is the delay allowed without extending the overall project completion deadline.
Option (C): Incorrect. Critical path has zero float, not negative float under normal schedules.
Option (D): Incorrect. Crashing refers to time-cost trade-off analysis.

HIGH-YIELD EXAM TAKEAWAY: Total Float ($LS - ES$) = delay without affecting project completion. Free Float = delay without affecting successor start.

Question 96 10.3 Project Planning & Scheduling: PERT

In Project Evaluation and Review Technique (PERT), given an optimistic time estimate of $a = 4$ days, a most likely time estimate of $m = 7$ days, and a pessimistic time estimate of $b = 16$ days, what are the expected activity duration (t_e) and its variance (σ^2)?

✓ **Correct Answer: (A)** $t_e = 8 \text{ days}, \sigma^2 = 4$

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

PERT Statistical Formulas (based on Beta distribution):

1. **Expected Activity Duration (t_e)**:

$$t_e = \frac{a + 4m + b}{6}$$

Substituting $a = 4, m = 7, b = 16$:

$$t_e = \frac{4 + 4(7) + 16}{6} = \frac{4 + 28 + 16}{6} = \frac{48}{6} = \mathbf{8} \text{ days}$$

2. **Standard Deviation (σ)**:

$$\sigma = \frac{b - a}{6} = \frac{16 - 4}{6} = \frac{12}{6} = 2 \text{ days}$$

3. **Variance (σ^2)**:

$$\sigma^2 = 2^2 = \mathbf{4}$$

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. $t_e = \frac{4+28+16}{6} = 8$ days, and $\sigma^2 = \left(\frac{16-4}{6}\right)^2 = 2^2 = 4$.

Option (B): Incorrect. 2 is standard deviation, not variance.

Option (C): Incorrect. Assumes expected time equals most likely time $m = 7$.

Option (D): Incorrect. Calculation error in weighting.

HIGH-YIELD EXAM TAKEAWAY: PERT formulas: Expected time $t_e = \frac{a+4m+b}{6}$; Variance $\sigma^2 = \left(\frac{b-a}{6}\right)^2$.

Question 97 10.4 Project Management: Procurement & Tendering in Nepal

Under Nepal's Public Procurement Act (PPA 2063) and Public Procurement Regulations (PPR 2064), what percentage of the estimated bid amount is typically required to be furnished by bidders as **Bid Security (Earnest Money)?**

✓ Correct Answer: **(B) 2.0% to 3.0% of the estimated bid amount**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Procurement Securities in Nepal (PPA 2063 / PPR 2064):
1. **Bid Security (Earnest Money Deposit - EMD)**: Required during tender submission to guarantee that the bidder will not withdraw their bid during the validity period and will sign the contract if awarded. As per Nepalese public procurement rules, Bid Security is fixed between **2% and 3%** of the official estimated project cost (valid for 30 days beyond bid validity).
2. **Performance Security**: Submitted by the winning bidder upon signing the contract, typically **5%** of the contract price (up to 10% for foreign contractors or unbalanced bids).
3. **Retention Money**: Deducted at 5% from running bills, released after the Defects Liability Period (DLP).

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. 0.5% - 1.0% is too low and does not meet the statutory threshold.
Option (B): Correct. Nepal PPA/PPR stipulates Bid Security between 2% and 3% of the estimated cost.
Option (C): Incorrect. 10% exceeds standard bid security; performance bond can reach 10% on unbalanced bids.
Option (D): Incorrect. Mobilization advance guarantees are separate bank guarantees.

HIGH-YIELD EXAM TAKEAWAY: Under Nepal Public Procurement Act: Bid Security $= 2\% - 3\%$ of estimate. Performance Security $= 5\%$ of contract.

Question 98 10.4 Project Management: Contract Administration

In engineering contract administration, what is the primary purpose of a **Liquidated Damages (LD) clause?**

✓ Correct Answer: **(B) To provide pre-agreed, legally enforceable compensatory damages to the employer for financial losses caused by contractor delays in completing the project beyond the contracted completion date**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Liquidated Damages in Engineering Contracts:
- Liquidated Damages (LD) represent a genuine pre-estimate of financial loss agreed upon by both parties at contract signing, payable by the contractor if they fail to complete the works by the contractual completion date.
- Under FIDIC and Nepalese standard bidding documents (PPMO):
- LD is assessed on a daily basis (typically **0.05% of the contract price per day of delay**).
- The maximum ceiling (cap) for liquidated damages is standardly set at **10% of the contract price**.
- LD is compensatory, not punitive (courts do not enforce purely punitive penalties). Once the 10% cap is reached, the employer has the right to terminate the contract.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Contract law prohibits punitive penalty clauses; LD must be compensatory.
Option (B): Correct. Pre-estimated compensation paid to the employer for contractor-caused project delays.
Option (C): Incorrect. Inflation adjustments are handled by Price Adjustment formulas.
Option (D): Incorrect. LD has no bearing on client QA responsibilities.

HIGH-YIELD EXAM TAKEAWAY: Liquidated Damages compensate the employer for delay (standard: 0.05% per day, capped at 10% of contract value).

Question 99 10.5 Engineering Professional Practice: Ethics

In international professional codes of ethics and the Nepal Engineering Council (NEC) Code of Conduct, what is universally established as the **paramount duty of a professional engineer?**

✓ Correct Answer: **(B) Holding paramount the safety, health, and welfare of the public and environmental sustainability in the execution of engineering work**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Fundamental Canon of Engineering Ethics:
The first and most fundamental canon of engineering ethics across all major statutory and professional bodies (Nepal Engineering Council, IEEE, NSPE, ICE):
> **"Engineers shall hold paramount the safety, health, and welfare of the public in the performance of their professional duties."**
This ethical duty overrides loyalty to employers, commercial profit motives, project schedules, and client preferences. When public safety or environmental integrity is compromised, engineers are ethically and legally bound to report the hazard and refuse to approve substandard designs.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Incorrect. Commercial profit is secondary to public safety.
Option (B): Correct. Public health, safety, and welfare is the paramount ethical responsibility of all engineers.
Option (C): Incorrect. Bypassing inspections to meet deadlines is a gross violation of professional ethics.
Option (D): Incorrect. Compliance with legal and statutory regulations is mandatory.

HIGH-YIELD EXAM TAKEAWAY: The primary canon of engineering ethics: Hold paramount the health, safety, and welfare of the public.

Question 100 10.6 Engineering Regulatory Body: Nepal Engineering Council (NEC)

Under the Nepal Engineering Council (NEC) Act 2055 and its amendments, which of the following statements regarding the licensure and registration of engineers in Nepal is strictly TRUE?

✓ Correct Answer: **(A) An engineering graduate holding a recognized Bachelor's degree in engineering who passes the council examination is registered under the 'General Registered Engineer' category**

ENGINEERING RATIONALE & TECHNICAL DERIVATION:

Nepal Engineering Council (NEC) Act 2055 & Regulations 2056:
1. The Nepal Engineering Council is an autonomous statutory regulatory body established in 2055 B.S. to regulate the engineering profession in Nepal.
2. **Registration Categories**:
- **General Registered Engineer (साधारण दर्तावाल इन्जिनियर)**: Granted to individuals holding a recognized Bachelor's degree in engineering from an accredited university who successfully pass the council's licensure examination.
- **Professional Registered Engineer (व्यावसायिक दर्तावाल इन्जिनियर)**: Requires minimum post-graduate degree or extensive verified professional experience (typically 5+ years) and passing a rigorous peer assessment.
- **Foreign Registered Engineer**: Temporary registration issued to foreign specialists working on specific authorized development projects in Nepal.
3. Practicing engineering or certifying designs without valid NEC registration is an offense punishable by law.

OPTION ANALYSIS & DISTRACTOR DIAGNOSTICS:

Option (A): Correct. Bachelor's graduates passing the licensure exam are registered as 'General Registered Engineer'.
Option (B): Incorrect. Professional Engineer requires advanced post-graduate qualifications, extensive experience, and assessment.
Option (C): Incorrect. Foreign engineers can obtain temporary registration for designated projects.
Option (D): Incorrect. NEC registration is strictly mandatory by law to practice engineering or certify designs in Nepal.

HIGH-YIELD EXAM TAKEAWAY: Under NEC Act 2055: Bachelor's degree + licensure exam = General Registered Engineer. Practicing without NEC registration is illegal.