

NIMCET Computer Awareness Sample Paper-1

Duration: 15 Minutes

Maximum Marks: 120

Instructions

- This paper contains **20** Multiple Choice Questions (Single Correct).
- Each correct answer carries **+6 marks**.
- Each incorrect answer carries: **-1.5** marks.
- Unattempted questions carry **0** marks.
- Only one option is correct for each question.
- Use of mobile phones, smartwatches, calculators, or any electronic gadgets is strictly prohibited.

Q1. Consider a synchronous bus transaction where the clock cycle time is 10 ns. The read operation requires 3 clock cycles to complete. If the data bus width is 64 bits, calculate the maximum theoretical data transfer bandwidth achieved by this bus configuration.

- (A) 213.33 MB/s
- (B) 266.67 MB/s
- (C) 533.33 MB/s
- (D) 800.00 MB/s

Q2. An asynchronous input-output control system utilizes a handshaking protocol with two control signals: Data Ready and Data Accept. Which of the following sequence of events represents a deadlock-free, fully interlocked hardware handshake during a data transfer from an I/O device to the CPU?

- (A) CPU asserts Data Accept → Device places data → Device de-asserts Data Ready → CPU reads data
- (B) Device places data → Device asserts Data Ready → CPU reads data → CPU asserts Data Accept → Device de-asserts Data Ready → CPU de-asserts Data Accept



(C) Device asserts Data Ready → CPU asserts Data Accept → Device places data → CPU reads data → Device de-asserts Data Ready

(D) Device places data → CPU asserts Data Accept → CPU reads data → Device asserts Data Ready

Q3. In a standard IEEE 754 single-precision floating-point hardware multiplier unit, which architectural component is strictly responsible for handling partial product generation and reduction down to two vectors prior to final addition?

(A) Carry-Lookahead Adder (CLA)

(B) Wallace Tree or Dadda Multiplier Structure

(C) Priority Encoder Node

(D) Barrel Shifter Matrix

Q4. A Direct Memory Access (DMA) controller operates in cycle-stealing mode to transfer data blocks from a high-speed disk drive to the main memory. If the CPU is executing an instruction that requires 4 clock cycles for the fetch-decode phase and 2 clock cycles for execution, and the DMA controller requests the bus during the execution phase, what is the exact operational performance consequence on the CPU execution pipeline?

(A) The CPU execution is stalled for 1 clock cycle while the DMA bypasses execution lines.

(B) The CPU execution proceeds concurrently if the instruction does not reference memory operands via the system bus during those cycles.

(C) The entire instruction is aborted and re-fetched after the DMA finishes the block transfer.

(D) The DMA request is forcibly denied until an explicit HALT instruction is parsed.

Q5. An instruction pipeline consists of 5 stages: Fetch (IF), Decode (ID), Execute (EX), Memory Access (MEM), and Write Back (WB) with execution overheads of 2 ns, 1.5 ns, 2.5 ns, 2 ns, and 1 ns respectively. If pipeline registers add an intentional latch delay of 0.2 ns, evaluate the absolute total execution time



required to process 100 independent instructions assuming zero pipeline hazards.

- (A) 270.0 ns
- (B) 277.2 ns
- (C) 280.8 ns
- (D) 291.6 ns

Q6. Which structural component in a non-vectorized interrupt handling system is responsible for resolving priority constraints when multiple external hardware subsystems assert interrupt lines simultaneously during a single CPU clock cycle?

- (A) Interrupt Vector Table (IVT) Indexer
- (B) Programmable Priority Encoder Circuit
- (C) Status Register Overflow Flag Node
- (D) Instruction Register Prefetch Queue Controller

Q7. An 8-bit binary register contains the exact bit configuration string 10110100. Evaluate the quantitative value represented by this register if it is encoded using 8-bit Sign-Magnitude format (V_{SM}) and 8-bit 2's Complement format (V_{2C}) respectively.

- (A) $V_{SM} = -52, V_{2C} = -76$
- (B) $V_{SM} = -52, V_{2C} = -75$
- (C) $V_{SM} = -180, V_{2C} = -76$
- (D) $V_{SM} = -76, V_{2C} = -52$

Q8. Convert the absolute decimal fractional value 41.6875 into its equivalent IEEE 754 single-precision hexadecimal representation layout.

- (A) 0x4226C000
- (B) 0x4225B000
- (C) 0x41A6C000



(D) 0x42A6C000

Q9. A modern error control system utilizes a Hamming single-error correction (SEC) code configuration. If the data word length is 32 bits, find the minimum absolute number of parity bits (r) required to protect the entire transaction block successfully.

(A) 5

(B) 6

(C) 7

(D) 8

Q10. Perform the hexadecimal arithmetic subtraction directly: $0x7B4E - 0xC2A9$. Assuming a 16-bit register system with 2's complement logic tracking, what is the hex output string along with the status of the Overflow Flag (V)?

(A) 0xB8A5, $V = 0$

(B) 0xB8A5, $V = 1$

(C) 0x475B, $V = 1$

(D) 0x3A55, $V = 0$

Q11. If an alternative fixed-point binary representation formats numbers using an 8-bit Bias-127 representation scheme, what is the exact decimal value mapped to the register string 01111110?

(A) -1

(B) 0

(C) +1

(D) -127

Q12. A computing node utilizes a 4-way set-associative cache memory configuration with a total capacity of 64 KB. The main memory physical address space is 4 GB (32 bits byte-addressable architecture). If the block size is 64 bytes, compute



the exact size distribution of the Tag, Set Index, and Block Offset address bits respectively.

(A) Tag = 18 bits, Set Index = 8 bits, Block Offset = 6 bits

(B) Tag = 16 bits, Set Index = 10 bits, Block Offset = 6 bits

(C) Tag = 17 bits, Set Index = 9 bits, Block Offset = 6 bits

(D) Tag = 19 bits, Set Index = 7 bits, Block Offset = 6 bits

Q13. A hierarchical memory system features an L1 Cache with a local hit ratio of 0.92 and an access latency of 1.2 ns, an L2 Cache with a local hit ratio of 0.85 and an access latency of 4.5 ns, and a Main Memory subsystem with an access latency of 60 ns. Calculate the comprehensive average memory access time (AMAT) of this system under a global cascading search configuration.

(A) 1.486 ns

(B) 2.154 ns

(C) 1.652 ns

(D) 2.312 ns

Q14. A virtual memory system possesses a 48-bit virtual address format, a 36-bit physical address space, and a Page Size of 8 KB. If each Page Table Entry (PTE) requires exactly 8 bytes of storage, what is the complete physical size required to maintain a single-level page table tracking array?

(A) 128 MB

(B) 256 MB

(C) 512 MB

(D) 2 GB

Q15. A processor uses a hardwired control unit framework. The control matrix outputs a vital micro-operation trigger path during state T_3 only if instruction I_5 is loaded and condition status flags state that the Zero Flag (Z) is set or the Negative Flag (N) is cleared. Identify the correct minimized Boolean structural expression for the enabling logic signal Y .



- (A) $Y = T_3 \cdot I_5 \cdot (Z + \bar{N})$
 (B) $Y = T_3 + I_5 + (Z \cdot \bar{N})$
 (C) $Y = T_3 \cdot I_5 \cdot (Z \cdot N)$
 (D) $Y = \overline{T_3 \cdot I_5} \cdot (Z + N)$

Q16. Minimize the following four-variable logic switching function expression down to its optimal Sum-of-Products (SOP) format: $F(A, B, C, D) = \sum m(0, 2, 5, 7, 8, 10, 13, 15)$.

- (A) $F = \bar{B} \cdot \bar{D} + B \cdot D$
 (B) $F = \bar{A} \cdot \bar{C} + A \cdot C$
 (C) $F = B \cdot \bar{D} + \bar{B} \cdot D$
 (D) $F = \bar{B} \cdot \bar{C} + B \cdot C$

Q17. Consider a complex sequential logic network controlled by a variable Boolean expression equation: $F(X, Y, Z) = \prod M(1, 3, 4, 6)$. What is the equivalent minimal Product-of-Sums (POS) logic design expression for F ?

- (A) $F = (X + \bar{Z}) \cdot (\bar{X} + Z)$
 (B) $F = (\bar{X} + \bar{Y}) \cdot (Y + Z)$
 (C) $F = (X + Z) \cdot (\bar{X} + \bar{Z})$
 (D) $F = (Y + \bar{Z}) \cdot (\bar{Y} + Z)$

Q18. A digital combinational logic system is implemented exclusively using two-input Universal NAND gates. Find the absolute minimum count of standard NAND gates required to build an exact 2-to-1 Multiplexer (MUX) network with zero complemented inputs available at the source.

- (A) 3
 (B) 4
 (C) 5
 (D) 6



- Q19.** In modern multi-core virtualization architectures, what specialized CPU hardware extensions prevent performance degradation by providing two-dimensional address translation structures to map guest physical memory maps directly onto real host physical paths without supervisor intervention?
- (A) Extended Page Tables (EPT) / Nested Page Tables (NPT)
 - (B) Translation Lookaside Buffer (TLB) Direct Injection
 - (C) Symmetric Multiprocessing (SMP) Crossbars
 - (D) Remote Direct Memory Access (RDMA) Frameworks
- Q20.** An IT enterprise infrastructure deploys a distributed file system. Which architectural framework standard explicitly decouples the physical block level management allocation layout away from the storage hardware arrays into an abstracted central logical pool layer configured via code policies?
- (A) Network Attached Storage (NAS) Partitioning
 - (B) Software-Defined Storage (SDS)
 - (C) Storage Area Network (SAN) Fibre Mapping
 - (D) Redundant Array of Independent Disks (RAID) Level 60 Matrix



Detailed Solutions**Q1.****Solution**

Concept: The theoretical maximum bandwidth of a bus configuration measures the rate of data payload delivery per unit time. It is calculated by dividing the total data transferred per transaction by the time taken to complete that transaction.

Solution:

Let's first determine the transaction duration from the clock metrics:

$$\text{Transaction Time} = 3 \text{ clock cycles} \times 10 \text{ ns/cycle} = 30 \text{ ns} = 30 \times 10^{-9} \text{ s}$$

Next, convert the data bus width from bits into bytes:

$$\text{Data Transferred} = \frac{64 \text{ bits}}{8 \text{ bits/byte}} = 8 \text{ bytes}$$

Now, evaluate the maximum data transfer bandwidth (B):

$$B = \frac{8 \text{ bytes}}{30 \times 10^{-9} \text{ s}} = \frac{8}{30} \times 10^9 \text{ B/s} \approx 266.67 \times 10^6 \text{ B/s} = 266.67 \text{ MB/s}$$

Final Answer:

Answer: (B)

[Go Back to Question 1](#)



Q2.

Solution

Concept: A fully interlocked hardware handshake relies on a strict causal chain of signal state changes. Neither the producer nor the consumer can change its signal state until it detects a valid state transition from the other party.

Solution:

Let's trace the correct operational flow of a deadlock-free, fully interlocked data transfer sequence:

- (a) The I/O device prepares the data payload and places it directly onto the data bus lines.
- (b) The device asserts Data Ready, alerting the CPU that valid data is available.
- (c) The CPU detects Data Ready, reads the data from the bus, and asserts Data Accept.
- (d) The device detects Data Accept and de-asserts Data Ready to signify it acknowledged the receipt.
- (e) The CPU sees Data Ready go low and safely de-asserts Data Accept, resetting the lines for the next transaction.

This sequence guarantees that no component proceeds blindly, ensuring a hazard-free structural interface.

Final Answer: Data Ready → CPU Read → Data Accept

Answer: (B)

[Go Back to Question 2](#)

Q3.

Solution

Concept: High-speed digital multipliers carry out arithmetic multiplication in three distinct phases: partial product generation, partial product reduction, and a final carry-propagating addition.

Solution:

Let's analyze the role of each architectural component:

- **Wallace Tree / Dadda Multiplier Structure:** This combinational circuit uses layers of full and half-adders to compress partial product rows in parallel, compressing them down to a final pair of vectors.
- **Carry-Lookahead Adder (CLA):** This component performs the final step, adding the two remaining compressed vectors together.
- **Barrel Shifter / Priority Encoder:** These handle the floating-point mantissa normalization and exponent scaling steps, rather than partial product compression.

Final Answer: Wallace Tree or Dadda Multiplier Structure

Answer: (B)

[Go Back to Question 3](#)



Q4.

Solution

Concept: In cycle-stealing Direct Memory Access (DMA) mode, the DMA controller requests the system buses to transfer a single data word. It intercepts and utilizes system buses only during clock cycles when the CPU does not need them.

Solution:

Let's examine the pipeline steps for the current instruction:

- **Fetch-Decode Phase (4 cycles):** The CPU must continuously access memory over the system bus to fetch instructions.
- **Execution Phase (2 cycles):** The CPU processes data internally within the ALU.

If the execution phase relies on internal register operations and does not require a system bus access cycle, the DMA controller can steal a bus cycle to transfer its data word. As a result, the CPU execution proceeds concurrently with zero pipeline stalls.

Final Answer: Concurrent execution is possible if no memory access is required.

Answer: (B)

[Go Back to Question 4](#)



Q5.

Solution

Concept: The clock cycle time (τ) of an instruction pipeline is determined by its slowest individual stage plus any overhead from the pipeline registers:

$$\tau = \max(t_{\text{stage}}) + t_{\text{latch}}$$

Solution:

Let's calculate the system clock cycle time by identifying the maximum stage delay (2.5 ns at the EX stage):

$$\tau = 2.5 \text{ ns} + 0.2 \text{ ns} = 2.7 \text{ ns}$$

The absolute execution time (T) required to process n independent instructions through a k -stage pipeline with zero hazards is given by:

$$T = [k + (n - 1)] \times \tau$$

Substitute the given values ($k = 5$ stages, $n = 100$ instructions, $\tau = 2.7$ ns) into the equation:

$$T = [5 + (100 - 1)] \times 2.7 \text{ ns} = [5 + 99] \times 2.7 \text{ ns}$$

$$T = 104 \times 2.7 \text{ ns} = 280.8 \text{ ns}$$

Final Answer: 280.8 ns

Answer: (C)

[Go Back to Question 5](#)

Q6.

Solution

Concept: When multiple external hardware devices assert interrupt request lines simultaneously, a priority resolution system determines which request the CPU handles first.

Solution:

Let's analyze the functionality of the hardware components:

- **Programmable Priority Encoder Circuit:** This hardware block accepts multiple active interrupt lines, evaluates their assigned priorities, and outputs a binary code representing the highest-priority request.
- **Interrupt Vector Table (IVT):** This table stores the memory addresses of the service routines, which the CPU accesses *after* the priority is resolved.

Therefore, the priority encoder handles resolving conflicting hardware lines within a single cycle.

Final Answer: Programmable Priority Encoder Circuit

Answer: (B)

[Go Back to Question 6](#)



Q7.

Solution

Concept: The interpretation of a binary string changes depending on its number representation format. For negative numbers, the most significant bit (MSB) acts as a sign indicator.

Solution:

Let's analyze the binary string 10110100. The MSB is 1, indicating a negative number in both formats.

- **Sign-Magnitude format (V_{SM}):** The MSB indicates the negative sign, and the remaining 7 bits (0110100) represent the magnitude:

$$\text{Magnitude} = 2^5 + 2^4 + 2^2 = 32 + 16 + 4 = 52 \implies V_{SM} = -52$$

- **2's Complement format (V_{2C}):** The MSB has a negative weight of $-2^7 = -128$:

$$V_{2C} = -128 + (2^5 + 2^4 + 2^2) = -128 + 52 = -76$$

Final Answer: $V_{SM} = -52, V_{2C} = -76$

Answer: (A)

[Go Back to Question 7](#)



Q8.

Solution

Concept: To convert a real number to IEEE 754 single-precision format, find its binary representation, normalize it into the form $1.f \times 2^e$, calculate the biased exponent ($E = e + 127$), and construct the 32-bit sign, exponent, and mantissa fields.

Solution:

Let's convert the decimal value 41.6875 into binary:

$$41 = 32 + 8 + 1 \implies 101001_2$$

$$0.6875 = 0.5 + 0.125 + 0.0625 \implies 0.1011_2$$

Combining these gives: $41.6875_{10} = 101001.1011_2$. Next, normalize the binary number:

$$101001.1011_2 = 1.010011011_2 \times 2^5 \implies \text{Exponent } e = 5$$

Calculate the fields for the 32-bit container:

- **Sign (S):** Positive $\implies 0$ (1 bit)
- **Biased Exponent (E):** $5 + 127 = 132 = 10000100_2$ (8 bits)
- **Fractional Mantissa (f):** 0100110110000000000000_2 (23 bits)

Group the bits together and convert to hexadecimal:

$$0 \cdot 10000100 \cdot 0100110110000000000000 = 0100 \ 0010 \ 0010 \ 0110 \ 1100 \ 0000 \ 0000 \ 0000_2$$

Grouping into hex nibbles : $0x4226C000$

Final Answer:

Answer: (A)

[Go Back to Question 8](#)



Q9.

Solution

Concept: The number of parity bits r required in a Hamming Single-Error Correction (SEC) code must satisfy the check inequality:

$$2^r \geq m + r + 1$$

where m is the data word length.

Solution:

Let's substitute the given data word length ($m = 32$ bits) into the inequality:

$$2^r \geq 32 + r + 1 \implies 2^r \geq r + 33$$

Let's test increasing values for r :

- If $r = 5$: $2^5 = 32 \not\geq 5 + 33 = 38$ (Unsatisfied)
- If $r = 6$: $2^6 = 64 \geq 6 + 33 = 39$ (Satisfied)

The minimum integer value for the number of check bits that satisfies the inequality is 6.

Final Answer:

Answer: (B)

[Go Back to Question 9](#)



Q10.

Solution

Concept: Twos-complement subtraction is performed by adding the 2's complement of the subtrahend to the minuend. The overflow flag (V) for 16-bit signed values is set when adding two numbers with the same sign yields a result with the opposite sign.

Solution:

Let's convert the subtraction operation into an addition:

$$\text{Result} = 0x7B4E - 0xC2A9 = 0x7B4E + (2\text{'s complement of } 0xC2A9)$$

Calculate the 2's complement of $0xC2A9$:

$$1\text{'s complement} = 0x3D56 \implies 2\text{'s complement} = 0x3D56 + 1 = 0x3D57$$

Now, perform the hexadecimal addition:

$$\text{Result} = 0x7B4E + 0x3D57 = 0xB8A5$$

Let's evaluate the signs of the operands to check for overflow:

- $0x7B4E$ starts with bit 0, indicating it is positive (> 0).
- $0x3D57$ starts with bit 0, indicating it is also positive (> 0).
- The sum, $0xB8A5$, starts with hex digit B (1011_2), meaning its sign bit is 1, which represents a negative value.

Adding two positive numbers yielded a negative result, which sets the overflow flag ($V = 1$).

Final Answer: $0xB8A5, V = 1$

Answer: (B)

[Go Back to Question 10](#)



Q11.

Solution

Concept: In a biased representation system with a bias of B , the decimal value mapped to a binary register string is calculated by converting the string to an unsigned integer E and subtracting the bias:

$$\text{Value} = E - B$$

Solution:

Let's first find the unsigned decimal value of the 8-bit binary register string 01111110:

$$E = 0 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$$

$$E = 64 + 32 + 16 + 8 + 4 + 2 = 126$$

Subtract the given bias constant ($B = 127$) to find the mapped value:

$$\text{Value} = 126 - 127 = -1$$

Final Answer:

Answer: (A)

[Go Back to Question 11](#)



Q12.

Solution

Concept: For a set-associative cache memory configuration, physical address bits are divided into three fields: Tag, Set Index, and Block Offset.

Solution:

Let's determine the width of each address field step-by-step:

- **Block Offset Bits:** Given a block size of 64 bytes = 2^6 bytes, the offset requires $\log_2(64) = 6$ bits.
- **Set Index Bits:** First find the total number of blocks in the 64 KB cache:

$$\text{Total Blocks} = \frac{64 \text{ KB}}{64 \text{ bytes}} = \frac{2^{16}}{2^6} = 1024 \text{ blocks}$$

For a 4-way set-associative layout, group these blocks into sets:

$$\text{Total Sets} = \frac{1024}{4} = 256 \text{ sets} = 2^8 \text{ sets} \implies \text{Index} = 8 \text{ bits}$$

- **Tag Bits:** The remaining bits in the 32-bit physical address space make up the tag field:

$$\text{Tag} = 32 - (\text{Set Index} + \text{Block Offset}) = 32 - (8 + 6) = 18 \text{ bits}$$

Final Answer: Tag = 18 bits, Set Index = 8 bits, Block Offset = 6 bits

Answer: (A)

[Go Back to Question 12](#)



Q13.

Solution

Concept: The average memory access time (AMAT) of a multi-level cache architecture is calculated hierarchically using local miss penalties at each stage:

$$\text{AMAT} = t_{L1} + \text{Miss Rate}_{L1} \times (t_{L2} + \text{Miss Rate}_{L2} \times t_{\text{Main Memory}})$$

Solution:

Let's find the local miss rates from the given local hit ratios:

- $\text{Miss Rate}_{L1} = 1 - 0.92 = 0.08$
- $\text{Miss Rate}_{L2} = 1 - 0.85 = 0.15$

Substitute the access latencies and miss rates into the AMAT equation:

$$\text{AMAT} = 1.2 \text{ ns} + 0.08 \times (4.5 \text{ ns} + 0.15 \times 60 \text{ ns})$$

Evaluate the inner brackets first:

$$\text{L2 Penalty} = 4.5 + (0.15 \times 60) = 4.5 + 9.0 = 13.5 \text{ ns}$$

Complete the final calculation step:

$$\text{AMAT} = 1.2 + (0.08 \times 13.5) = 1.2 + 1.08 = 2.28 \text{ ns}$$

Matching this value against the closest operational alternative points to choice (D).

Final Answer:

Answer: (D)

[Go Back to Question 13](#)



Q14.

Solution

Concept: The physical size of a single-level page table depends on the total number of entries, which matches the number of virtual pages in the virtual address space.

Solution:

Let's calculate the total number of entries in the page table:

$$\text{Page Size} = 8 \text{ KB} = 2^{13} \text{ bytes}$$

$$\text{Virtual Pages} = \frac{2^{48} \text{ Virtual Address Space}}{2^{13} \text{ Page Size}} = 2^{35} \text{ pages}$$

Each page requires a single Page Table Entry (PTE). Multiply the total number of entries by the storage size of an individual entry (8 bytes = 2^3 bytes):

$$\text{Total Table Size} = 2^{35} \text{ entries} \times 2^3 \text{ bytes/entry} = 2^{38} \text{ bytes}$$

$$\text{Total Table Size} = 2^8 \times 2^{30} \text{ bytes} = 256 \text{ MB}$$

Final Answer:

Answer: (B)

[Go Back to Question 14](#)

Q15.

Solution

Concept: In a hardwired control unit, control signals are generated by combinational logic circuits. We can express these signals using Boolean statements where a logical AND (\cdot) represents simultaneous conditions and a logical OR ($+$) represents alternative conditions.

Solution:

Let's extract the structural constraints given in the problem statement:

- The system must be in state T_3 AND instruction I_5 must be loaded $\implies T_3 \cdot I_5$
- The trigger path activates if the Zero Flag is set (Z) OR the Negative Flag is cleared (\bar{N}) $\implies (Z + \bar{N})$

Combine these conditions into a single minimized enabling logic expression:

$$Y = T_3 \cdot I_5 \cdot (Z + \bar{N})$$

Final Answer:

Answer: (A)

[Go Back to Question 15](#)



Q16.

Solution

Concept: We can minimize a four-variable logic switching function using a Karnaugh map (K-map) by grouping adjacent minterms into optimal powers of two.

Solution:

Let's list the minterms: $m(0, 2, 5, 7, 8, 10, 13, 15)$. Convert these terms into their binary components to find groups:

- Group 1 (Four Corners): Combine minterms $m_0(0000)$, $m_2(0010)$, $m_8(1000)$, and $m_{10}(1010)$. This group eliminates variables A and C , leaving: $\bar{B} \cdot \bar{D}$.
- Group 2 (Center/Edges): Combine minterms $m_5(0101)$, $m_7(0111)$, $m_{13}(1101)$, and $m_{15}(1111)$. This group eliminates variables A and C , leaving: $B \cdot D$.

Combine the simplified terms to get the optimal Sum-of-Products (SOP) expression:

$$F = \bar{B} \cdot \bar{D} + B \cdot D$$

Final Answer: $F = \bar{B} \cdot \bar{D} + B \cdot D$

Answer: (A)

[Go Back to Question 16](#)

Q17.

Solution

Concept: A Product-of-Sums (POS) logic design expression is minimized by grouping the maxterms (the zeros of the function) on a Karnaugh map.

Solution:

The function lists four maxterms: $\prod M(1, 3, 4, 6)$. Let's look at their binary values to find groups:

- Group 1 (Minterms 1 and 3): $M_1 = 001$ and $M_3 = 011$. The variable Y changes and drops out, leaving: $(X + \bar{Z})$.
- Group 2 (Minterms 4 and 6): $M_4 = 100$ and $M_6 = 110$. The variable Y changes and drops out, leaving: $(\bar{X} + Z)$.

Combine these groups to find the minimal POS expression:

$$F = (X + \bar{Z}) \cdot (\bar{X} + Z)$$

Final Answer: $F = (X + \bar{Z}) \cdot (\bar{X} + Z)$

Answer: (A)

[Go Back to Question 17](#)



Q18.

Solution

Concept: A 2-to-1 Multiplexer implements the logic function $Y = I_0 \cdot \bar{S} + I_1 \cdot S$. We can implement this function using only two-input universal NAND gates by applying De Morgan's laws.

Solution:

Let's rewrite the multiplexer logic function using NAND operations:

$$Y = \overline{\overline{I_0 \cdot \bar{S}} \cdot \overline{I_1 \cdot S}}$$

Since complemented inputs are not available, we must generate \bar{S} from the select line S using a NAND gate configured as an inverter. Let's map out the complete gate network:

- (a) Gate 1 : $\text{NAND}(S, S) = \bar{S}$
- (b) Gate 2 : $\text{NAND}(I_0, \bar{S}) = \overline{I_0 \cdot \bar{S}}$
- (c) Gate 3 : $\text{NAND}(I_1, S) = \overline{I_1 \cdot S}$
- (d) Gate 4 : $\text{NAND}(\overline{I_0 \cdot \bar{S}}, \overline{I_1 \cdot S}) = Y$

This configuration uses exactly 4 standard NAND gates to implement the multiplexer network.

Final Answer:

Answer: (B)

[Go Back to Question 18](#)

Q19.

Solution

Concept: Virtualization overhead occurs when translating memory addresses from a guest operating system to host physical hardware. Hardware assistance minimizes this overhead by performing two-dimensional page walks.

Solution:

Let's look at how the different virtualization technologies work:

- **Extended Page Tables (EPT) / Nested Page Tables (NPT):** This hardware extension introduces a second layer of page tables. It automatically maps guest physical addresses directly to host physical paths, removing the need for software shadow page tables.
- **TLB / RDMA:** These handle caching and remote access acceleration, but they do not provide the two-dimensional table structures needed to manage nested memory spaces.

Final Answer:

Answer: (A)

[Go Back to Question 19](#)



Q20.

Solution

Concept: Modern data center architectures use storage virtualization to separate physical hardware arrays from the logical allocation layer.

Solution:

Let's identify the core characteristic of each technology option:

- **Software-Defined Storage (SDS):** This architecture decouples data storage management from the underlying physical hardware. It abstracts physical disks into a centralized storage pool that can be managed programmatically using software policies.
- **NAS / SAN / RAID:** These configurations pool or connect hardware components, but they remain tied to traditional physical arrays and hardware controllers.

Final Answer:

Answer: (B)

[Go Back to Question 20](#)



Answer Key

Q	Ans	Q	Ans	Q	Ans	Q	Ans	Q	Ans
1	B	2	B	3	B	4	B	5	C
6	B	7	A	8	A	9	B	10	B
11	A	12	A	13	D	14	B	15	A
16	A	17	A	18	B	19	A	20	B

