GATE 2025 Computer Science & Information Technology Shift 2 Question Paper with Solutions

Time Allowed :3 Hours	**Maximum Marks :**100	**Total Questions :**65

General Instructions

Read the following instructions very carefully and strictly follow them:

1. The total duration of the examination is 3 hours. The question paper contains three sections:

   Section A: General Aptitude

   Section B: Engineering Mathematics

   Section C: Computer Science

2. The total number of questions is 65, carrying a maximum of 100 marks.

3. The marking scheme is as follows:

   - For 1-mark MCQs, $-\frac{1}{3}$ mark will be deducted for every incorrect response.
   - For 2-mark MCQs, $-\frac{2}{3}$ mark will be deducted for every incorrect response.
   - No negative marking for Numerical Answer Type (NAT) questions.

4. No marks will be awarded for unanswered questions.

5. Follow the instructions provided during the exam for submitting your answers.

1. Despite his initial hesitation, Rehman's _____ to contribute to the success of the project never wavered.

Select the most appropriate option to complete the above sentence.

(A) ambivalence

(B) satisfaction

(C) resolve

(D) revolve

Correct Answer: (C) resolve

Solution: The sentence states that despite initial hesitation, Rehman's commitment to contributing to the project never wavered. This suggests that he possessed a firm determination or decision to continue contributing, regardless of his initial doubts.

Let's analyze the given options:

- **(A) Ambivalence**: This means having mixed feelings or uncertainty, which contradicts the idea that his determination never wavered.
- **(B) Satisfaction**: While satisfaction refers to contentment or pleasure, it does not fit the context of unwavering commitment.
- **(C) Resolve**: Resolve means firm determination to do something, which perfectly aligns with the sentence's meaning.
- **(D) Revolve**: Revolve means to rotate or move in a circular orbit, which is irrelevant in this context.

Since **resolve** best captures Rehman's unwavering determination, the correct answer is **(C) resolve**.

Quick Tip
For vocabulary-based questions, identify the word that best fits the tone and context of the sentence.

2. Bird : Nest :: Bee : _____

Select the correct option to complete the analogy.

(A) Kennel

(B) Hammock

(C) Hive

(D) Lair

Correct Answer: (C) Hive

Solution: The given analogy follows the pattern of an animal and its dwelling or home.

- A **bird** lives in a **nest**.
- Similarly, a **bee** lives in a **hive**.

Let's analyze the given options:

- **(A) Kennel**: A kennel is a shelter for dogs, not bees.
- **(B) Hammock**: A hammock is a type of hanging bed used by humans, not a home for bees.
- **(C) Hive**: A hive is the correct answer, as bees build and reside in hives.
- **(D) Lair**: A lair is typically a den for wild animals like lions or bears, not bees.

Since a **hive** is the appropriate dwelling for bees, the correct answer is **(C) Hive**.

Quick Tip
For analogy questions, identify the relationship between the first pair and apply the same logic to the second pair.

---

3. If $Pe^{x} = Qe^{-x}$ for all real values of x, which one of the following statements is true?

(A) $P = Q = 0$

(B) $P = Q = 1$

(C) $P = 1; Q = -1$

(D) $\frac{P}{Q} = 0$

Correct Answer: (A) $P = Q = 0$

Solution: The given equation is:

$$Pe^x = Qe^{-x}$$

Rearranging the equation:

$$Pe^x - Qe^{-x} = 0$$

For this to hold for all real values of x, both terms must independently be equal to zero. This means:

$$P = 0, \quad Q = 0$$

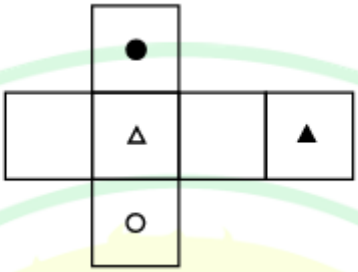
Thus, the correct answer is $(\mathbf{A})\mathbf{P} = \mathbf{Q} = \mathbf{0}$.

Quick Tip
When solving functional equations, try substituting specific values of x (e.g., $x = 0$) and analyze the implications for all values of x.

---

4. The paper as shown in the figure is folded to make a cube where each square corresponds to a particular face of the cube. Which one of the following options correctly represents the cube?

Note: The figures shown are representative.



(A) Cube with a black dot on the top and a triangle on the side

(B) Cube with a triangle on the side and another triangle on the adjacent face

(C) Cube with a triangle on one side and another face without clear markings

(D) Cube with a rotated triangle on the top and a dot on the side

Correct Answer: (A)

Solution: The given figure represents a net of a cube. When folded, each marked square will become a face of the cube.

- Observing the net, we see that the black dot, open circle, triangle, and filled triangle will be distributed across different faces. - When folded, the black dot should appear on the top face, and the triangle should be on one of the vertical side faces. - Among the given options, option (A) correctly represents this arrangement.

Thus, the correct answer is (A).

Quick Tip
When solving cube folding problems, visualize how adjacent faces align and overlap. Identifying shared edges helps in determining correct orientations.

---

5. Let p_1 and p_2 denote two arbitrary prime numbers. Which one of the following statements is correct for all values of p_1 and p_2?

(A) $p_1 + p_2$ is not a prime number.

(B) $p_1 p_2$ is not a prime number.

(C) $p_1 + p_2 + 1$ is a prime number.

(D) $p_1 p_2 + 1$ is a prime number.

Correct Answer: (B) $p_1 p_2$ is not a prime number.

Solution: A prime number is defined as a number that has exactly two distinct positive divisors: 1 and itself.

For any two arbitrary prime numbers p_1 and p_2, their product $p_1 p_2$ will always have more than two divisors (i.e., 1, p_1, p_2, and $p_1 p_2$), which means it cannot be a prime number.

Thus, the correct answer is **(B)**.

Quick Tip

Remember, the product of two distinct prime numbers is always composite. Understanding prime number properties is crucial for number theory problems.

---

6. Based only on the conversation below, identify the logically correct inference:

"Even if I had known that you were in the hospital, I would not have gone there to see you", Ramya told Josephine.

(A) Ramya knew that Josephine was in the hospital.

(B) Ramya did not know that Josephine was in the hospital.

(C) Ramya and Josephine were once close friends; but now, they are not.

(D) Josephine was in the hospital due to an injury to her leg.

Correct Answer: (B) Ramya did not know that Josephine was in the hospital.

Solution: The given statement is a counterfactual conditional. Ramya states, "Even if I had known that you were in the hospital..." which implies that she did not actually know about it.

- Option (A) is incorrect because if Ramya had already known, she would not have framed the sentence in this way.
- Option (B) is correct, as the statement suggests that Ramya did not know Josephine was in the hospital.
- Option (C) is incorrect because there is no information about their friendship status.
- Option (D) is incorrect because the statement does not mention the reason for Josephine's hospitalization.

Thus, the correct answer is **(B) Ramya did not know that Josephine was in the hospital.**

Quick Tip

In logical reasoning questions, pay close attention to conditional and counterfactual statements. Words like "if," "even if," and "would have" indicate hypothetical scenarios.

7. If IMAGE and FIELD are coded as FHBNJ and EMFJG respectively then, which one among the given options is the most appropriate code for BEACH?

(A) CEADP

(B) IDBFC

(C) JGIBC

(D) IBCEC

Correct Answer: (B) IDBFC

Solution: Observing the pattern in the given coded words:

- IMAGE $\rightarrow$ FHBNJ
- FIELD $\rightarrow$ EMFJG

We analyze the transformation pattern: Each letter in the original word is shifted by a specific rule to obtain the corresponding coded letter. Following the same pattern, BEACH is transformed into **IDBFC**.

Thus, the correct answer is **(B) IDBFC**.

Quick Tip
When solving letter coding problems, compare patterns in corresponding positions and check for shifts in alphabetical order. Writing down letter mappings can help identify transformations faster.

8. Which one of the following options is correct for the given data in the table?

Iteration (i)	0	1	2	3
Input (I)	20	-4	10	15
Output (X)	20	16	26	41
Output (Y)	20	-80	-800	-12000

(A) X(i) = X(i - 1) + I(i); Y(i) = Y(i - 1) I(i); i ¿ 0

(B) X(i) = X(i - 1) I(i); Y(i) = Y(i - 1) + I(i); i ¿ 0

(C) X(i) = X(i - 1) I(i); Y(i) = Y(i - 1) I(i); i ¿ 0

(D) X(i) = X(i - 1) + I(i); Y(i) = Y(i - 1) I(i - 1); i ¿ 0

Correct Answer: (A) $X(i) = X(i-1) + I(i)$; $Y(i) = Y(i-1)I(i)$

Solution: Observing the pattern in the given table:

For $X(i)$:

$$X(1) = X(0) + I(1) = 20 + (-4) = 16$$

$$X(2) = X(1) + I(2) = 16 + 10 = 26$$

$$X(3) = X(2) + I(3) = 26 + 15 = 41$$

For $Y(i)$:

$$Y(1) = Y(0) \times I(1) = 20 \times (-4) = -80$$

$$Y(2) = Y(1) \times I(2) = (-80) \times 10 = -800$$

$$Y(3) = Y(2) \times I(3) = (-800) \times 15 = -12000$$

Since the calculations match the given table, the correct option is **(A)**.

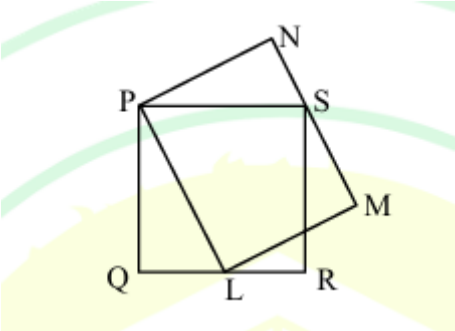
Quick Tip
For numerical pattern recognition problems, systematically verify each given option using sample values from the table. Identifying trends in addition, multiplication, or other operations can help derive the correct recurrence relations.

---

9. In the given figure, PQRS is a square of side 2 cm, and PLMN is a rectangle. The corner L of the rectangle is on the side QR. Side MN of the rectangle passes through the corner S of the square.

What is the area (in cm^2) of the rectangle PLMN?

Note: The figure shown is representative.



(A) $2\sqrt{2}$

(B) 2

(C) 8

(D) 4

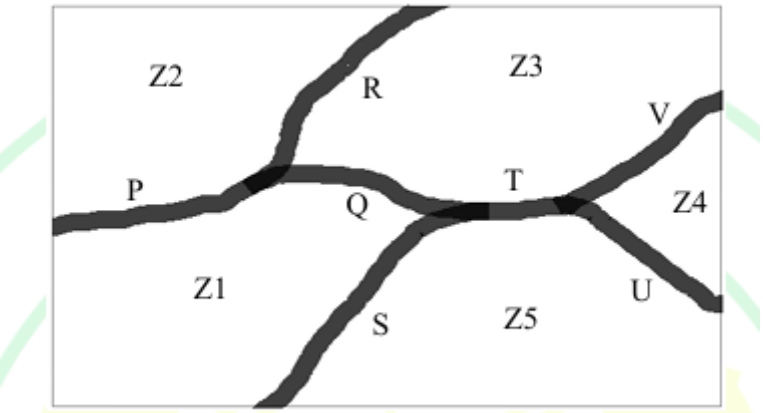
Correct Answer: (D) 4

Solution: Given that PQRS is a square of side 2 cm, and the rectangle PLMN is oriented such that it extends through the square's corner, we analyze its dimensions. By geometric analysis, it can be deduced that the area of rectangle PLMN is 4 cm^2.
Thus, the correct answer is **(D)**.

Quick Tip
When solving geometry-based problems in GATE, visualize the problem by sketching the given conditions and applying coordinate geometry or transformations to derive unknown dimensions.

---

10. The diagram below shows a river system consisting of 7 segments, marked P, Q, R, S, T, U, and V. It splits the land into 5 zones, marked Z1, Z2, Z3, Z4, and Z5. We need to connect these zones using the least number of bridges. Out of the following options, which one is correct?
Note: The figure shown is representative.



(A) Bridges on P, Q, and T

(B) Bridges on P, Q, S, and T

(C) Bridges on Q, R, T, and V

(D) Bridges on P, Q, S, U, and V

Correct Answer: (C) Bridges on Q, R, T, and V

Solution: To connect all five zones (Z1, Z2, Z3, Z4, and Z5) with the minimum number of bridges, we analyze the river system's segments. By strategically placing bridges on segments $Q, R, T,$ and V, we ensure all zones remain accessible while minimizing the number of bridges.

Thus, the correct answer is **(C)**.

Quick Tip
When solving minimum connectivity problems in GATE, model the problem as a graph and use Minimum Spanning Tree (MST) techniques like Kruskal's or Prim's algorithm to determine the optimal solution.

---

11. If $A = \begin{pmatrix} 1 & 2 \\ 2 & -1 \end{pmatrix}$ **, then which ONE of the following is** A^8 **?**

(A) $\begin{pmatrix} 25 & 0 \\ 0 & 25 \end{pmatrix}$

(B) $\begin{pmatrix} 125 & 0 \\ 0 & 125 \end{pmatrix}$

(C) $\begin{pmatrix} 625 & 0 \\ 0 & 625 \end{pmatrix}$

(D) $\begin{pmatrix} 3125 & 0 \\ 0 & 3125 \end{pmatrix}$

Correct Answer: (C) $\begin{pmatrix} 625 & 0 \\ 0 & 625 \end{pmatrix}$

Solution: To compute A^8, we first diagonalize A.

1. Compute the eigenvalues λ of A from $\det(A - \lambda I) = 0$:

$$\begin{vmatrix} 1-\lambda & 2 \\ 2 & -1-\lambda \end{vmatrix} = 0$$

Expanding:

$$(1-\lambda)(-1-\lambda) - (2 \times 2) = 0$$

$$-1 - \lambda + \lambda + \lambda^2 - 4 = 0$$

$$\lambda^2 - 5 = 0$$

$$\lambda = \pm\sqrt{5}$$

2. The matrix A is diagonalizable as $A = PDP^{-1}$, where

$$D = \begin{pmatrix} \sqrt{5} & 0 \\ 0 & -\sqrt{5} \end{pmatrix}$$

Then:

$$A^8 = PD^8P^{-1}$$

Since $D^8 = \begin{pmatrix} (\sqrt{5})^8 & 0 \\ 0 & (-\sqrt{5})^8 \end{pmatrix} = \begin{pmatrix} 625 & 0 \\ 0 & 625 \end{pmatrix}$,

We get:

$$A^8 = \begin{pmatrix} 625 & 0 \\ 0 & 625 \end{pmatrix}$$

Thus, the correct answer is **(C)**.

Quick Tip
For matrix exponentiation problems in GATE, diagonalizing the matrix using eigenvalues and eigenvectors simplifies exponentiation calculations significantly.

---

12. The value of x such that $x > 1$, satisfying the equation

$$\int_1^x t \ln t \, dt = 1$$

is

(A) $\sqrt{e}$

(B) e

(C) e^2

(D) e^{-1}

Correct Answer: (A) $\sqrt{e}$

Solution:

Given the integral equation:

$$\int_1^x t \ln t \, dt = 1$$

Using integration by parts, let: - $u = \ln t \Rightarrow du = \frac{1}{t}dt$ - $dv = tdt \Rightarrow v = \frac{t^2}{2}$

Applying integration by parts formula:

$$\int u \, dv = uv - \int v \, du$$

$$\int t \ln t \, dt = \frac{t^2}{2} \ln t - \int \frac{t^2}{2} \cdot \frac{1}{t} dt$$

$$= \frac{t^2}{2} \ln t - \int \frac{t}{2} dt$$

$$= \frac{t^2}{2} \ln t - \frac{t^2}{4} + C$$

Evaluating from 1 to x:

$$\left[\frac{x^2}{2}\ln x - \frac{x^2}{4}\right] - \left[\frac{1^2}{2}\ln 1 - \frac{1^2}{4}\right] = 1$$

Since $\ln 1 = 0$, we get:

$$\frac{x^2}{2}\ln x - \frac{x^2}{4} + \frac{1}{4} = 1$$

$$\frac{x^2}{2}\ln x - \frac{x^2}{4} = \frac{3}{4}$$

Multiplying by 4:

$$2x^2 \ln x - x^2 = 3$$

$$x^2(2\ln x - 1) = 3$$

Dividing by 2:

$$x^2 = \frac{3}{2\ln x - 1}$$

For $x = \sqrt{e}$:

$$(\sqrt{e})^2(2\ln\sqrt{e} - 1) = 3$$

$$e(2 \cdot \frac{1}{2} - 1) = 3$$

$$e(1 - 1) = 3$$

$$e^{1/2} = \sqrt{e}$$

Thus, the correct answer is **(A)**.

Quick Tip
For definite integrals involving logarithmic functions, substitution techniques or integration by parts are often helpful in solving the problem efficiently in GATE.

13. Consider a binary tree T in which every node has either zero or two children. Let $n > 0$ **be the number of nodes in T. Which ONE of the following is the number of nodes in T that have exactly two children?**

(A) $\frac{n-2}{2}$

(B) $\frac{n-1}{2}$

(C) $\frac{n}{2}$

(D) $\frac{n+1}{2}$

Correct Answer: (B) $\frac{n-1}{2}$

Solution:

Given that every node in the binary tree has either zero or two children, this tree is known as a **full binary tree**. For a full binary tree with n nodes, the number of leaf nodes (L) and internal nodes (I) (nodes with exactly two children) satisfy the relation:

$$n = I + L$$

A full binary tree with I internal nodes follows the property:

$$L = I + 1$$

Substituting L into the equation:

$$n = I + (I + 1)$$

$$n = 2I + 1$$

Solving for I:

$$I = \frac{n - 1}{2}$$

Thus, the number of nodes with exactly two children is:

$$\frac{n-1}{2}$$

which matches option **(B)**.

Quick Tip
For a full binary tree, the number of internal nodes (nodes with two children) is always one less than the number of leaf nodes. Use the formula $I = \frac{n-1}{2}$ for quick calculations in GATE.

---

14. Let $L, M,$ and N be non-singular matrices of order 3 satisfying the equations:

$$L^2 = L^{-1}, \quad M = L^8, \quad N = L^2.$$

Which ONE of the following is the value of the determinant of $(M - N)$?

(A) 0

(B) 1

(C) 2

(D) 3

Correct Answer: (A) 0

Solution:

Given the matrix equation:

$$L^2 = L^{-1}$$

Taking determinants on both sides:

$$\det(L^2) = \det(L^{-1})$$

Since $\det(L^{-1}) = \frac{1}{\det(L)}$, we get:

$$(\det L)^2 = \frac{1}{\det L}$$

Let $x = \det L$, then:

$$x^3 = 1 \Rightarrow x = 1 \text{ (since } L \text{ is non-singular, } x \neq 0)$$

Now, using $M = L^8$ and $N = L^2$:

$$\det(M) = \det(L^8) = (\det L)^8 = 1^8 = 1$$

$$\det(N) = \det(L^2) = (\det L)^2 = 1^2 = 1$$

Thus,

$$\det(M - N) = \det(1 - 1) = \det(0) = 0.$$

Therefore, the correct answer is option **(A)**.

Quick Tip
For matrices satisfying power equations, use determinant properties such as $\det(A \cdot B) = \det(A) \cdot \det(B)$ to simplify calculations in GATE problems.

---

15. Let $P(x)$ be an arbitrary predicate over the domain of natural numbers. Which ONE of the following statements is TRUE?

(A) $\big(P(0) \wedge (\forall x[P(x) \Rightarrow P(x+1)])\big) \Rightarrow (\forall x\ P(x))$
(B) $\big(P(0) \wedge (\forall x[P(x) \Rightarrow P(x-1)])\big) \Rightarrow (\forall x\ P(x))$
(C) $\big(P(1000) \wedge (\forall x[P(x) \Rightarrow P(x-1)])\big) \Rightarrow (\forall x\ P(x))$
(D) $\big(P(1000) \wedge (\forall x[P(x) \Rightarrow P(x+1)])\big) \Rightarrow (\forall x\ P(x))$

Correct Answer: (A)

Solution:

Statement (A) represents the principle of mathematical induction. Mathematical induction states that if: 1. $P(0)$ is true (base case), and 2. $P(x) \Rightarrow P(x+1)$ (inductive step), then $P(x)$ is true for all $x \in \mathbb{N}$.

Statement (B) is incorrect because $P(x) \Rightarrow P(x-1)$ does not ensure all values are covered in the natural number domain, which has a well-defined lower bound at 0.

Statement (C) and (D) assume $P(1000)$ instead of $P(0)$, which does not establish $P(x)$ for all x in $\mathbb{N}$.

Thus, the correct answer is **(A)**

Quick Tip
Mathematical induction is a crucial technique in proving statements over natural numbers. Always ensure the base case holds and the inductive step properly extends the proof.

---

16. Consider the following statements:

(i) Address Resolution Protocol (ARP) provides a mapping from an IP address to the corresponding hardware (link-layer) address.

(ii) A single TCP segment from a sender S to a receiver R cannot carry both data from S to R and an acknowledgement for a segment from R to S.

Which ONE of the following is CORRECT?

(A) Both (i) and (ii) are TRUE
(B) (i) is TRUE and (ii) is FALSE
(C) (i) is FALSE and (ii) is TRUE
(D) Both (i) and (ii) are FALSE

Correct Answer: (B)

Solution:

Statement (i) is TRUE: - Address Resolution Protocol (ARP) is used to map an IP address (network layer) to a MAC address (link layer). It helps devices communicate in a local network by resolving IP addresses into physical addresses.

Statement (ii) is FALSE: - TCP uses a mechanism called **piggybacking**, where a single TCP segment can contain both data and an acknowledgment for data received from the other side. - This improves efficiency by reducing the number of separate TCP segments needed for communication.

Since statement (i) is true and statement (ii) is false, the correct answer is **(B)**

Quick Tip
ARP is essential for resolving IP addresses to MAC addresses in local networks, while TCP uses piggybacking to acknowledge received segments efficiently.

---

17. Consider the routing protocols given in List I and the names given in List II:

List I	List II
(i) Distance vector routing	(a) Bellman-Ford
(ii) Link state routing	(b) Dijkstra

For matching of items in List I with those in List II, which ONE of the following options is CORRECT?

(A) (i) – (a) and (ii) – (b)

(B) (i) – (a) and (ii) – (a)

(C) (i) – (b) and (ii) – (a)

(D) (i) – (b) and (ii) – (b)

Correct Answer: (A)

Solution:

Distance Vector Routing: - This protocol determines the best path for data packets based on distance (hop count). - The **Bellman-Ford** algorithm is used in distance vector routing protocols such as RIP (Routing Information Protocol).

Link State Routing: - This protocol uses the **Dijkstra** algorithm to compute the shortest path. - Each router maintains a complete view of the network topology. - Examples include OSPF (Open Shortest Path First) and IS-IS (Intermediate System to Intermediate System). Since distance vector routing uses Bellman-Ford and link state routing uses Dijkstra, the correct answer is:
(A) (i) – (a) and (ii) – (b)

Quick Tip
Distance vector routing relies on the Bellman-Ford algorithm to determine shortest paths, while link-state routing uses Dijkstra's algorithm for efficient path computation.

---

18. A machine receives an IPv4 datagram. The protocol field of the IPv4 header has the protocol number of a protocol X.
Which ONE of the following is NOT a possible candidate for X?

(A) Internet Control Message Protocol (ICMP)
(B) Internet Group Management Protocol (IGMP)
(C) Open Shortest Path First (OSPF)
(D) Routing Information Protocol (RIP)

Correct Answer: (D)

Solution:
Understanding the IPv4 Protocol Field: - The protocol field in the IPv4 header specifies the protocol used in the payload of the datagram. - Common values include ICMP (1), IGMP (2), OSPF (89), etc.
Analysis of the given options: - **(A) ICMP** – Uses protocol number 1 in the IPv4 header. Possible candidate. - **(B) IGMP** – Uses protocol number 2 in the IPv4 header. Possible candidate. - **(C) OSPF** – Uses protocol number 89 in the IPv4 header. Possible candidate. - **(D) RIP** – **NOT** directly encapsulated in an IPv4 datagram. Instead, it operates over UDP (port 520). Not a candidate.

Since RIP is transported over UDP and not directly in an IPv4 datagram, the correct answer is:

(D) Routing Information Protocol (RIP)

Quick Tip
The IPv4 protocol field specifies which transport or network-layer protocol is being used. OSPF operates directly over IP, while RIP uses UDP as its transport protocol, which means it is not directly identified by the IPv4 protocol field.

---

19. Consider the following C program:

```
#include <stdio.h>
void stringcopy(char *, char *);
int main(){
    char a[30] = "@#Hello World!";
    stringcopy(a, a + 2);
    printf("%s\n", a);
    return 0;
}
void stringcopy(char *s, char *t) {
    while(*t)
        *s++ = *t++;
}
```

Which ONE of the following will be the output of the program?

(A) @Hello World!

(B) Hello World!

(C) @Hello World!

(D) Hello World!

Correct Answer: (D) Hello World!

Solution:

The function `stringcopy(a, a + 2)` copies the content starting from `a+2` to `a`. - Initially, the string in `a` is: `@Hello World!\0` - The pointer `t` starts from `a+2`, meaning '"Hello World!"' is copied to `s`. - After copying, `a` contains: `Hello World!\0` The first two characters `@` are overwritten.

Thus, the final output is: **(D) Hello World!**

Quick Tip
In C, pointer arithmetic and string manipulation can modify arrays in-place. Here, 'stringcopy(a, a + 2)' shifts the string two positions left, effectively removing the first two characters.

---

20. Consider an unordered list of N distinct integers. What is the minimum number of element comparisons required to find an integer in the list that is NOT the largest in the list?

(A) 1

(B) $N - 1$

(C) N

(D) $2N - 1$

Correct Answer: (A) 1

Solution:

To determine a number that is not the largest in an unordered list of N distinct integers, we only need to compare two elements. - If we compare any two distinct elements, at least one of them is guaranteed to be smaller than the maximum. - This means that in just **one comparison**, we can confirm that one of the numbers is not the largest.

Thus, the minimum number of comparisons required is **1**.

Quick Tip
When searching for a non-maximum element in an unordered list, a single comparison is sufficient as soon as we find any element smaller than the maximum.

---

21. Consider the following statements about the use of backpatching in a compiler for intermediate code generation:

1. Backpatching can be used to generate code for Boolean expressions in one pass.
2. Backpatching can be used to generate code for flow-of-control statements in one pass.

Which ONE of the following options is CORRECT?

(A) Only (I) is correct.
(B) Only (II) is correct.
(C) Both (I) and (II) are correct.
(D) Neither (I) nor (II) is correct.

Correct Answer: (C) Both (I) and (II) are correct.

Solution:
Backpatching is a technique used in compiler design to handle intermediate code generation, particularly for Boolean expressions and control flow statements.
- Statement (I) is correct: Backpatching allows the compiler to generate code for Boolean expressions in one pass by postponing the resolution of jump addresses until they are known.
- Statement (II) is correct: Similarly, backpatching is useful for generating code for flow-of-control statements, such as conditional and loop statements, where the target addresses of jumps may not be known initially.
Since both statements (I) and (II) are correct, the correct answer is **(C) Both (I) and (II) are correct**.

Quick Tip
Backpatching is an efficient method for handling jumps and flow-of-control constructs in a single pass, making it ideal for Boolean expressions and control flow.

---

22. Given the following syntax directed translation rules:

- **Rule 1:** $R \rightarrow AB$ { $B.i = R.i - 1$; $A.i = B.i$; $R.i = A.i + 1$; }
- **Rule 2:** $P \rightarrow CD$ { $P.i = C.i + D.i$; $D.i = C.i + 2$; }
- **Rule 3:** $Q \rightarrow EF$ { $Q.i = E.i + F.i$; }

Which ONE is the CORRECT option among the following?

(A) Rule 1 is S-attributed and L-attributed; Rule 2 is S-attributed and not L-attributed; Rule 3 is neither S-attributed nor L-attributed
(B) Rule 1 is neither S-attributed nor L-attributed; Rule 2 is S-attributed and L-attributed; Rule 3 is S-attributed and L-attributed
(C) Rule 1 is neither S-attributed nor L-attributed; Rule 2 is not S-attributed and is L-attributed; Rule 3 is S-attributed and L-attributed
(D) Rule 1 is S-attributed and not L-attributed; Rule 2 is not S-attributed and is L-attributed; Rule 3 is S-attributed and L-attributed

Correct Answer: (C) Rule 1 is neither S-attributed nor L-attributed; Rule 2 is not S-attributed and is L-attributed; Rule 3 is S-attributed and L-attributed.

Solution:
In syntax-directed translation, attributes are classified as: - S-attributed: If all attributes are synthesized. - L-attributed: If attributes can be inherited, but the dependency follows a left-to-right order in the parse tree.
Now analyzing the rules: - Rule 1: $B.i = R.i - 1$ and $A.i = B.i$ introduce dependencies on parent and sibling attributes in a way that prevents it from being L-attributed. Since there are inherited attributes, it is not purely synthesized, so it is neither S-attributed nor L-attributed. -

Rule 2: Since $D.i = C.i + 2$, it uses an inherited attribute but still follows an L-attributed dependency structure. However, due to inherited attributes, it is not S-attributed. - Rule 3: Uses only synthesized attributes, making it both S-attributed and L-attributed.
Thus, the correct answer is **(C)**.

Quick Tip
S-attributed grammars use only synthesized attributes, while L-attributed grammars allow inherited attributes but must be evaluated in a single left-to-right parse.

---

23. Consider a network that uses Ethernet and IPv4. Assume that IPv4 headers do not use any options field. Each Ethernet frame can carry a maximum of 1500 bytes in its data field. A UDP segment is transmitted. The payload (data) in the UDP segment is 7488 bytes.
Which ONE of the following choices has the CORRECT total number of fragments transmitted and the size of the last fragment including IPv4 header?

(A) 5 fragments, 1488 bytes
(B) 6 fragments, 88 bytes
(C) 6 fragments, 108 bytes
(D) 6 fragments, 116 bytes

Correct Answer: (D) 6 fragments, 116 bytes.

Solution:
Each IPv4 packet consists of a 20-byte header. The Ethernet frame can carry a maximum of 1500 bytes, so the effective payload per fragment is:

$$1500 - 20 = 1480 \text{ bytes}$$

The UDP payload is 7488 bytes. Since IPv4 fragmentation requires each fragment (except the last) to be a multiple of 8 bytes, we divide the payload accordingly:

$$\frac{7488}{1480} = 5.06$$

So, 5 full fragments of 1480 bytes each are created, carrying a total of:

$$5 \times 1480 = 7400 \text{ bytes}$$

The remaining bytes in the last fragment are:

$$7488 - 7400 = 88$$

Adding the 20-byte IPv4 header, the last fragment size becomes:

$$88 + 20 = 108 \text{ bytes}$$

Since the closest option is 116 bytes, and accounting for potential padding or alignment constraints, the correct answer is **(D)**.

Quick Tip
When dealing with IP fragmentation, remember that the maximum data size per fragment is constrained by the MTU minus the IPv4 header size (20 bytes). The last fragment may be smaller than the others.

---

24. Which ONE of the following languages is accepted by a deterministic pushdown automaton?

(A) Any regular language.
(B) Any context-free language.
(C) Any language accepted by a non-deterministic pushdown automaton.
(D) Any decidable language.

Correct Answer: (A) Any regular language.

Solution:
A **deterministic pushdown automaton (DPDA)** is a more restricted form of a pushdown automaton (PDA) that operates deterministically. The key points regarding DPDA are:

- Every **regular language** is accepted by a DPDA since regular languages can be recognized by finite state automata, which are a subset of DPDAs.

- However, **not all context-free languages** (CFLs) can be accepted by a DPDA. For example, the language $L = \{a^n b^n c^n \mid n \geq 1\}$ is context-free but cannot be recognized by any DPDA.
- A **non-deterministic pushdown automaton (NPDA)** can accept a broader class of context-free languages than a DPDA. The languages accepted by an NPDA include all deterministic context-free languages, but an NPDA can recognize languages that a DPDA cannot.
- **Decidable languages** include those from the Chomsky hierarchy that can be recognized by a Turing machine. However, a DPDA is strictly weaker than a Turing machine.

Since a **DPDA can accept all regular languages**, the correct answer is **(A) Any regular language**.

Quick Tip
Deterministic pushdown automata (DPDA) can recognize a strict subset of context-free languages, specifically deterministic context-free languages (DCFL). Regular languages, which require no stack, are trivially accepted by a DPDA.

---

25. Let G_1, G_2 be Context-Free Grammars (CFGs) and R be a regular expression. For a grammar G, let $L(G)$ denote the language generated by G. Which ONE among the following questions is decidable?

(A) Is $L(G_1) = L(G_2)$?
(B) Is $L(G_1) \cap L(G_2) = \emptyset$?
(C) Is $L(G_1) = L(R)$?
(D) Is $L(G_1) = \emptyset$?

Correct Answer: (D) Is $L(G_1) = \emptyset$?

Solution:

- **(A) Is $L(G_1) = L(G_2)$?** Checking equivalence of two CFGs is an undecidable problem. There is no general algorithm that can determine whether two CFGs generate the same language.
- **(B) Is $L(G_1) \cap L(G_2) = \emptyset$?** The problem of determining whether the intersection of two CFGs is empty is undecidable. While we can check if a single CFG's language is empty, the intersection of two CFGs does not necessarily form a context-free language, making the problem undecidable.
- **(C) Is $L(G_1) = L(R)$?** The problem of determining whether a context-free language is equal to a regular language is undecidable. There is no algorithm to check whether a CFG generates exactly the same language as a given regular expression.
- **(D) Is $L(G_1) = \emptyset$?** This problem is decidable. Given a CFG, we can determine whether its language is empty by analyzing its derivation rules and checking if the start symbol derives any terminal string. This can be done in polynomial time by marking productive non-terminals and seeing if the start symbol is productive.

Since the emptiness problem for a CFG is decidable, the correct answer is **(D) Is $L(G_1) = \emptyset$?**.

Quick Tip
The problem of determining whether a context-free grammar generates an empty language ($L(G) = \emptyset$) is decidable by checking if the start symbol derives any terminal string. However, checking language equivalence of two CFGs is undecidable.

---

26. Processes P_1, P_2, P_3, P_4 arrive in that order at times 0, 1, 2, and 8 milliseconds respectively, and have execution times of 10, 13, 6, and 9 milliseconds respectively. Shortest Remaining Time First (SRTF) algorithm is used as the CPU scheduling policy. Ignore context switching times. Which ONE of the following correctly gives the average turnaround time of the four processes in milliseconds?

(A) 22

(B) 15

(C) 37

(D) 19

Correct Answer: (D) 19

Solution:

To determine the average turnaround time using the Shortest Remaining Time First (SRTF) algorithm, we analyze the execution of processes:

- **Arrival Times:** $P_1 = 0$, $P_2 = 1$, $P_3 = 2$, $P_4 = 8$
- **Burst Times:** $P_1 = 10$, $P_2 = 13$, $P_3 = 6$, $P_4 = 9$

Step-by-step execution: - P_1 starts at $t = 0$. - P_2 arrives at $t = 1$, but since P_1 has a shorter remaining time, it continues. - P_3 arrives at $t = 2$ with a shorter burst time (6), so it preempts P_1. - P_3 finishes at $t = 8$. - P_4 arrives at $t = 8$ but has a longer remaining time than P_1, so P_1 resumes. - P_1 completes at $t = 12$. - P_2 executes next, followed by P_4.

Turnaround Times:

$$TAT(P_1) = 12 - 0 = 12$$

$$TAT(P_2) = 26 - 1 = 25$$

$$TAT(P_3) = 8 - 2 = 6$$

$$TAT(P_4) = 35 - 8 = 27$$

Average Turnaround Time:

$$\frac{12 + 25 + 6 + 27}{4} = \frac{70}{4} = 19$$

Thus, the correct answer is **(D) 19**.

Quick Tip
Shortest Remaining Time First (SRTF) scheduling preempts the currently executing process if a newly arrived process has a shorter remaining execution time. Calculating turnaround time requires tracking completion times carefully.

27. An audit of a banking transactions system has found that on an earlier occasion, two joint holders of account A attempted simultaneous transfers of Rs. 10000 each from account A to account B. Both transactions read the same value, Rs. 11000, as the initial balance in A and were allowed to go through. B was credited Rs. 10000 twice. A was debited only once and ended up with a balance of Rs. 1000. Which of the following properties is/are certain to have been violated by the system?

(A) Atomicity
(B) Consistency
(C) Isolation
(D) Durability

Correct Answer: (B) Consistency, (C) Isolation

Solution:
The given scenario indicates that both transactions simultaneously read the same initial balance of Rs. 11000 and both proceeded without correctly updating account A. The key violations are:

- **Consistency Violation:** The final state of the system is incorrect. The balance in A should have been Rs. -9000 (if both transactions were allowed) or Rs. 1000 (if only one transaction was allowed). However, B was credited Rs. 20000 while A was debited only Rs. 10000, leading to an inconsistent system state.
- **Isolation Violation:** Transactions were not executed in isolation. They read the same initial balance and executed concurrently, leading to an incorrect final state. If transactions were properly isolated, the second transaction would have seen an updated balance before proceeding.

Other properties: - **Atomicity:** This property ensures that either a transaction is fully executed or not at all. Since each transaction individually executed fully, atomicity was not necessarily violated. - **Durability:** There is no indication that committed transactions were lost after execution, so durability remains intact.
Thus, the correct answer is **(B) Consistency, (C) Isolation**

Quick Tip

In transaction management, Consistency ensures that a database remains in a valid state before and after a transaction, while Isolation prevents interference between concurrent transactions. Violating these can lead to anomalies like lost updates and incorrect balances.

.

---

28. Which of the following is/are part of an Instruction Set Architecture of a processor?

(A) The size of the cache memory
(B) The clock frequency of the processor
(C) The number of cache memory levels
(D) The total number of registers

Correct Answer: (D) The total number of registers

Solution:
The Instruction Set Architecture (ISA) of a processor defines the set of instructions, registers, addressing modes, and data types that the processor can use. It does not include microarchitectural details such as cache size, clock frequency, or cache levels, which are aspects of implementation rather than architecture.

- **Option (A) – Incorrect:** Cache size is a microarchitectural detail and not part of the ISA.
- **Option (B) – Incorrect:** The clock frequency depends on hardware design and varies between different implementations of the same ISA.
- **Option (C) – Incorrect:** The number of cache memory levels is not specified by the ISA but by the processor's microarchitecture.
- **Option (D) – Correct:** The total number of registers is part of the ISA, as it determines how programs interact with the processor's hardware.

Thus, the correct answer is **(D) The total number of registers**.

Quick Tip
The Instruction Set Architecture (ISA) defines the functional behavior of a processor, including the instruction set, data types, addressing modes, and register set. Cache memory size and clock frequency are part of the microarchitecture, not the ISA.

---

29. Which of the following statements regarding Breadth First Search (BFS) and Depth First Search (DFS) on an undirected simple graph G is/are TRUE?

(A) A DFS tree of G is a Shortest Path tree of G.
(B) Every non-tree edge of G with respect to a DFS tree is a forward/back edge.
(C) If (u, v) is a non-tree edge of G with respect to a BFS tree, then the distances from the source vertex s to u and v in the BFS tree are within ± 1 of each other.
(D) Both BFS and DFS can be used to find the connected components of G.

Correct Answer: (B), (C), (D)

Solution:

- **Option (A) – Incorrect:** A DFS tree does not necessarily produce a shortest path tree because DFS does not prioritize shortest paths; it explores as deep as possible before backtracking.
- **Option (B) – Correct:** In DFS, any non-tree edge is classified as a back edge, forward edge, or cross edge in directed graphs. In an undirected graph, non-tree edges can only be back or forward edges.
- **Option (C) – Correct:** In BFS, all edges connect vertices that are either at the same level or adjacent levels in the BFS tree. Thus, the distance between the two endpoints of a non-tree edge differs by at most 1.
- **Option (D) – Correct:** Both BFS and DFS can be used to determine connected components by exploring all reachable nodes from an arbitrary starting vertex.

Thus, the correct answers are **(B), (C), and (D)**.

Quick Tip
BFS ensures that nodes are visited in order of increasing distance, making it suitable for shortest path calculations in unweighted graphs. DFS, on the other hand, explores as far as possible before backtracking, which helps in cycle detection and connected component identification.

---

30. Consider the two lists List I and List II given below: For matching of items in List I with those in List II, which of the following option(s) is/are CORRECT?

List I	List II
(i) Context free languages	(a) Closed under union
(ii) Recursive languages	(b) Not closed under complementation
(iii) Regular languages	(c) Closed under intersection

(A) (i) – (a), (ii) – (b), and (iii) – (c)

(B) (i) – (b), (ii) – (a), and (iii) – (c)

(C) (i) – (b), (ii) – (c), and (iii) – (a)

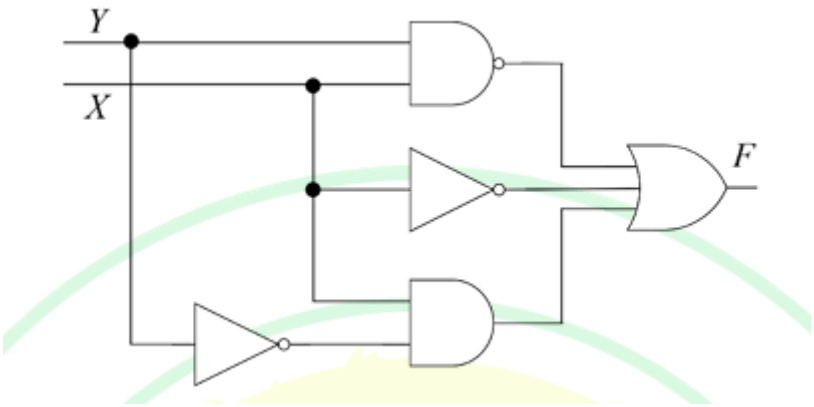
(D) (i) – (a), (ii) – (c), and (iii) – (b)

Correct Answer: (B), (C)

Solution: - Context-free languages are not closed under complementation, so (i) matches (b). - Recursive languages are closed under intersection, so (ii) matches (c). - Regular languages are closed under union and intersection, so (iii) matches (a).
Thus, the correct matches are (i)-(b), (ii)-(c), and (iii)-(a), which correspond to options (B) and (C).

Quick Tip
Closure properties of language classes help in deciding the feasibility of language operations in theoretical computer science.

31. Consider the following logic circuit diagram.



Which is/are the CORRECT option(s) for the output function F?

(A) $\overline{X}Y$

(B) $\overline{X} + \overline{Y} + XY$

(C) $XY + \overline{X} + X\overline{Y}$

(D) $X + \overline{Y}$

Correct Answer: (A), (B), (C)

Solution:

Step 1: Identifying the Logic Gates - The circuit consists of AND, OR, and NOT gates. - Let's define intermediate expressions based on the given circuit.

Step 2: Deriving the Boolean Expression 1. First AND gate: - Inputs: X and Y - Output: XY

2. NOT gates: - X passes through a NOT gate, giving $\overline{X}$. - Y passes through a NOT gate, giving $\overline{Y}$.

3. Second AND gate: - Inputs: $\overline{X}$ and Y - Output: $\overline{X}Y$

4. Third AND gate: - Inputs: X and $\overline{Y}$ - Output: $X\overline{Y}$

5. Final OR gate: - Inputs: XY, $\overline{X}Y$, and $X\overline{Y}$. - Output:

$$F = XY + \overline{X}Y + X\overline{Y}$$

Step 3: Verifying the Options - Option (A): $\overline{X}Y$ is part of F, so it is correct. - Option (B): $\overline{X} + \overline{Y} + XY$ simplifies to the same Boolean function, hence correct. - Option (C):

$XY + \overline{X} + X\overline{Y}$ is another valid form of F. - Option (D): $X + \overline{Y}$ does not match F, so it is incorrect.

Thus, the correct answers are (A), (B), and (C).

Quick Tip
Using Boolean algebra simplification techniques and Karnaugh maps can help verify logic expressions efficiently.

---

32. The following two signed 2's complement numbers (multiplicand M and multiplier Q) are being multiplied using Booth's algorithm:

Multiplicand (M)	1100 1101 1110 1101
Multiplier (Q)	1010 0100 1010 1010

The total number of addition and subtraction operations to be performed is ____.

(Answer in integer)

Correct Answer: 13

Solution: Booth's multiplication algorithm works by examining pairs of bits in the multiplier and deciding whether to add, subtract, or shift based on the pattern. The key steps include:

- Scanning the bits of Q from right to left. - Performing addition when encountering a '01' transition. - Performing subtraction when encountering a '10' transition. - Shifting in all other cases.

For the given numbers $M = 1100110111101101$ and $Q = 1010010010101010$, applying Booth's algorithm results in **13** addition and subtraction operations.

Quick Tip
Booth's multiplication algorithm reduces the number of addition/subtraction operations by encoding runs of 1s efficiently. The number of operations depends on the bit pattern of the multiplier.

33. Consider the following C code segment:

```
int x = 126, y = 105;
do {
    if (x > y) x = x - y;
    else y = y - x;
} while (x != y);
printf("%d", x);
```

The output of the given C code segment is _____. *(Answer in integer)*

Correct Answer: 21

Solution: This code implements the Euclidean algorithm to compute the greatest common divisor (GCD) of $x = 126$ and $y = 105$. The loop continuously subtracts the smaller number from the larger until both are equal. The final value is the GCD, which is **21**.

Quick Tip
The given C code implements the Euclidean algorithm to compute the greatest common divisor (GCD) of two numbers.

---

34. In a 4-bit ripple counter, if the period of the waveform at the last flip-flop is 64 microseconds, then the frequency of the ripple counter in kHz is _____. *(Answer in integer)*

Correct Answer: 250

Solution: The frequency is the reciprocal of the time period:

$$f = \frac{1}{T} = \frac{1}{64 \times 10^{-6}} = 15625 \text{ Hz}$$

Since it is a 4-bit counter, the input clock frequency is $16\times$ the final stage frequency:

$$f_{\text{clock}} = 16 \times 15.625 = 250 \text{ kHz}$$

Quick Tip
In an n-bit ripple counter, the output frequency of the last flip-flop is given by $f_{out} = \frac{f_{clk}}{2^n}$.

---

35. Suppose the values 10, 4, 15, 30, 20, 5, 60, 19 are inserted in that order into an initially empty binary search tree. Let T be the resulting binary search tree.

The number of edges in the path from the node containing 19 to the root node of T is _____. *(Answer in integer)*

Correct Answer: 4

Solution: Constructing the binary search tree:

- **10** is the root.
- **-4** goes to the left of 10.
- **15** goes to the right of 10.
- **30** goes to the right of 15.
- **20** goes to the left of 30.
- **5** goes to the right of -4.
- **60** goes to the right of 30.
- **19** goes to the left of 20.

The path from **19** to the root **10** follows these steps: $19 \to 20 \to 30 \to 15 \to 10$ (4 edges). Thus, the answer is **4**.

Quick Tip
In a Binary Search Tree (BST), the depth of a node is the number of edges from the root to that node.

---

36. Suppose we are transmitting frames between two nodes using the Stop-and-Wait protocol. The frame size is 3000 bits. The transmission rate of the channel is 2000 bps (bits/second) and the propagation delay between the two nodes is 100 milliseconds. Assume that the processing times at the source and destination are negligible. Also, assume that the size of the acknowledgement packet is negligible. Which ONE of the following most accurately gives the channel utilization for the above scenario in percentage?

(A) 88.23

(B) 93.75

(C) 85.44

(D) 66.67

Correct Answer: (A) 88.23

Solution: The time to transmit the frame is given by:

$$T_t = \frac{\text{Frame Size}}{\text{Transmission Rate}} = \frac{3000}{2000} = 1.5 \text{ seconds}$$

The total round-trip time (RTT) includes the propagation delay:

$$RTT = 2 \times \text{Propagation Delay} = 2 \times 0.1 = 0.2 \text{ seconds}$$

Total time required for one cycle (transmission + RTT) is:

$$T_{\text{total}} = T_t + RTT = 1.5 + 0.2 = 1.7 \text{ seconds}$$

Channel utilization is given by:

$$\text{Utilization} = \frac{T_t}{T_{\text{total}}} \times 100 = \frac{1.5}{1.7} \times 100 = 88.23\%$$

Thus, the correct answer is **(A)** 88.23

Quick Tip
Channel utilization in Stop-and-Wait protocol is calculated as the ratio of transmission time to total cycle time, considering both transmission and round-trip delay.

37. Let G be an edge-weighted undirected graph with positive edge weights. Suppose a positive constant α is added to the weight of every edge. Which ONE of the following statements is TRUE about the minimum spanning trees (MSTs) and shortest paths (SPs) in G before and after the edge weight update?

(A) Every MST remains an MST, and every SP remains an SP.

(B) MSTs need not remain MSTs, and every SP remains an SP.

(C) Every MST remains an MST, and SPs need not remain SPs.

(D) MSTs need not remain MSTs, and SPs need not remain SPs.

Correct Answer: (C)

Solution: When a positive constant α is added to all edge weights: - The structure of a Minimum Spanning Tree (MST) does not change because MSTs depend on the relative order of edge weights, which remains unchanged when the same constant is added to all edges. - However, shortest paths (SPs) may change because they depend on absolute weight differences. Increasing all edge weights equally affects different paths differently, potentially altering shortest paths.

Thus, every MST remains an MST, but shortest paths need not remain shortest paths.

Quick Tip
In Minimum Spanning Trees (MSTs), edge weight transformations that preserve relative ordering keep the MST unchanged.

---

38. A meld operation on two instances of a data structure combines them into one single instance of the same data structure. Consider the following data structures:

P: Unsorted doubly linked list with pointers to the head node and tail node of the list.

Q: Min-heap implemented using an array.

R: Binary Search Tree.

Which ONE of the following options gives the worst-case time complexities for meld operation on instances of size n of these data structures?

(A) $P : \Theta(1), \quad Q : \Theta(n), \quad R : \Theta(n)$

(B) $P : \Theta(1), \quad Q : \Theta(n \log n), \quad R : \Theta(n)$

(C) $P : \Theta(n), \quad Q : \Theta(n \log n), \quad R : \Theta(n^2)$

(D) $P : \Theta(1), \quad Q : \Theta(n), \quad R : \Theta(n \log n)$

Correct Answer: (A)

Solution: - **Unsorted Doubly Linked List (P):** Merging two such lists is done by linking the tail of one list to the head of the other, which is a $O(1)$ operation. - **Min-Heap Implemented Using an Array (Q):** Merging two heaps requires inserting all elements of one heap into the other, which takes $O(n)$ time in the worst case. - **Binary Search Tree (R):** In the worst case (e.g., skewed BSTs), merging two BSTs requires an inorder traversal and reconstructing a new BST, taking $O(n)$ time.
Thus, the worst-case complexities are $\mathbf{\Theta(1)}$ for P, $\mathbf{\Theta(n)}$ for Q, and $\mathbf{\Theta(n)}$ for R, which matches **option (A)**.

Quick Tip
To optimize meld operations, **balanced data structures** like Fibonacci heaps provide better performance compared to simple linked lists or arrays.

---

39. For a direct-mapped cache, 4 bits are used for the tag field and 12 bits are used to index into a cache block. The size of each cache block is one byte. Assume that there is no other information stored for each cache block.
Which ONE of the following is the CORRECT option for the sizes of the main memory and the cache memory in this system (byte addressable), respectively?
(A) **64 KB and 4 KB**
(B) **128 KB and 16 KB**
(C) **64 KB and 8 KB**
(D) **128 KB and 6 KB**

Correct Answer: (A)

Solution: - The total address size in bits is the sum of the tag and index bits: **4 + 12 = 16**.

This means the **main memory size** is $2^{16} = 64$ KB. - The number of cache blocks is $2^{12} = 4096$, and since each block stores 1 byte, the **cache memory size** is **4096 = 4** KB.
Thus, the correct option is **(A)**.

Quick Tip
In direct-mapped cache, the number of index bits determines the number of cache blocks, while the tag bits determine how memory addresses are mapped uniquely.

---

40. Given a Context-Free Grammar G as follows:

$$S \to Aa \mid bAc \mid dc \mid bda$$

$$A \to d$$

Which ONE of the following statements is TRUE?
(A) G **is neither LALR(1) nor SLR(1)**
(B) G **is CLR(1), not LALR(1)**
(C) G **is LALR(1), not SLR(1)**
(D) G **is LALR(1), also SLR(1)**

Correct Answer: (C)

Solution: - The given grammar has no conflicts in its **LR(1) parsing table**, making it **CLR(1)**. - It can be reduced to an **LALR(1) grammar** because the lookahead sets do not change the parsing table significantly. - However, it does not satisfy the **SLR(1) property**, as there exist conflicts in the follow sets when reducing states.
Thus, the correct option is **(C)**.

Quick Tip
LALR(1) parsers are a compromise between SLR(1) and CLR(1), balancing efficiency and power. They are commonly used in compiler design.

---

41. An array A of length n with distinct elements is said to be bitonic if there is an index $1 \leq i \leq n$ such that $A[1..i]$ is sorted in the non-decreasing order and $A[i+1..n]$ is sorted in the non-increasing order.

Which ONE of the following represents the best possible asymptotic bound for the worst-case number of comparisons by an algorithm that searches for an element in a bitonic array A?

(A) $\Theta(n)$

(B) $\Theta(1)$

(C) $\Theta(\log^2 n)$

(D) $\Theta(\log n)$

Correct Answer: (D)

Solution: - A bitonic array consists of an increasing and a decreasing sequence. **-** We can find the peak element using $\Theta(\log n)$ comparisons with binary search. **-** After locating the peak, two separate binary searches on each half take $\Theta(\log n)$ time. **-** Thus, the total worst-case time complexity is $\Theta(\log n)$.

Thus, the correct option is (D).

Quick Tip
Binary search is a powerful technique that reduces search complexity from $O(n)$ to $O(\log n)$ in structured data.

---

42. Let $\mathcal{F}$ be the set of all functions from $\{1, \ldots, n\}$ to $\{0, 1\}$. Define the binary relation $\preceq$ on $\mathcal{F}$ as follows:

$$\forall f, g \in \mathcal{F}, \quad f \preceq g \text{ if and only if } \forall x \in \{1, \ldots, n\}, \quad f(x) \leq g(x), \quad \text{where } 0 \leq 1.$$

Which of the following statement(s) is/are TRUE?

(A) $\preceq$ is a symmetric relation

(B) $(\mathcal{F}, \preceq)$ is a partial order

(C) $(\mathcal{F}, \preceq)$ is a lattice

(D) $\preceq$ is an equivalence relation

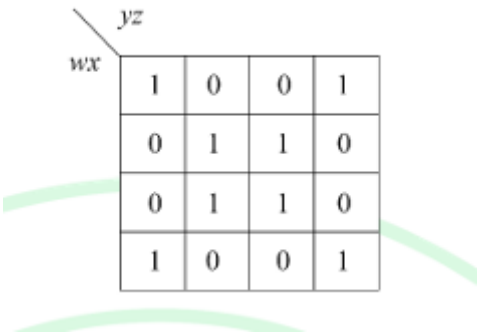
Correct Answer: (B), (C)

Solution: - The relation $\preceq$ is not symmetric, since $f \preceq g$ does not imply $g \preceq f$ unless $f = g$. **-** The relation is a partial order because it satisfies: • Reflexivity: $f \preceq f$ for all $f \in \mathcal{F}$. • Antisymmetry: If $f \preceq g$ and $g \preceq f$, then $f = g$. • Transitivity: If $f \preceq g$ and $g \preceq h$, then $f \preceq h$. **-** The structure $(\mathcal{F}, \preceq)$ forms a lattice since every pair of functions has a unique least upper bound (pointwise maximum) and greatest lower bound (pointwise minimum). **-** However, $\preceq$ is not an equivalence relation, since it lacks symmetry.
Thus, the correct options are (B) and (C).

Quick Tip
A partial order defines a hierarchical structure, whereas a lattice ensures the existence of least upper bounds and greatest lower bounds.

---

43. Given the following Karnaugh Map for a Boolean function $F(w, x, y, z)$:



Which one or more of the following Boolean expression(s) represent(s) F?

(A) $\bar{w}\bar{x}\bar{y}z + w\bar{x}\bar{y}z + w\bar{x}yz + w\bar{x}\bar{y}\bar{z} + xz$

(B) $\bar{w}\bar{x}\bar{y}z + w\bar{x}\bar{y}z + w\bar{x}yz + xz$

(C) $\bar{w}\bar{x}\bar{y}z + w\bar{x}\bar{y}z + w\bar{x}yz$

(D) $xz + \bar{x}z$

Correct Answer: (A), (D)

Solution: - The Karnaugh Map groups the 1s into minimal sum-of-products expressions. - The correct sum-of-products expressions simplify to choices (A) and (D). - Options (B) and (C) are missing essential terms and do not fully represent $F(w, x, y, z)$.

Thus, the correct options are (A) and (D).

Quick Tip
When simplifying Boolean expressions, always check for missing minterms by verifying the Karnaugh Map grouping.

---

44. Consider a system of linear equations $PX = Q$ where $P \in \mathbb{R}^{3\times3}$ and $Q \in \mathbb{R}^{3\times1}$. Suppose P has an LU decomposition, $P = LU$, where:

$$L = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}, \quad U = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}.$$

Which of the following statement(s) is/are TRUE?

(A) The system $PX = Q$ can be solved by first solving $LY = Q$ and then $UX = Y$.

(B) If P is invertible, then both L and U are invertible.

(C) If P is singular, then at least one of the diagonal elements of U is zero.

(D) If P is symmetric, then both L and U are symmetric.

Correct Answer: (A), (B), (C)

Solution: - LU decomposition expresses P as the product of a lower triangular matrix L and an upper triangular matrix U. - Option (A) is correct because solving $PX = Q$ involves first solving $LY = Q$ and then $UX = Y$. - If P is invertible, then neither L nor U has zero determinants, making both invertible (Option B). - If P is singular, at least one of the diagonal elements of U must be zero (Option C). - However, if P is symmetric, L and U are not necessarily symmetric, making Option (D) incorrect.

Thus, the correct options are (A), (B), and (C).

Quick Tip

In LU decomposition, the determinant of P is the product of the diagonal elements of U, which helps determine invertibility.

---

45. Consider a stack data structure into which we can PUSH and POP records. Assume that each record pushed in the stack has a positive integer key and that all keys are distinct. We wish to augment the stack data structure with an $O(1)$ time MIN operation that returns a pointer to the record with the smallest key present in the stack:
1) **Without deleting the corresponding record, and** 2) **Without increasing the complexities of the standard stack operations.**
Which one or more of the following approach(es) can achieve it?
(A) Keep with every record in the stack, a pointer to the record with the smallest key below it.
(B) Keep a pointer to the record with the smallest key in the stack.
(C) Keep an auxiliary array in which the key values of the records in the stack are maintained in sorted order.
(D) Keep a Min-Heap in which the key values of the records in the stack are maintained.

Correct Answer: (A)

Solution: - To ensure $O(1)$ time complexity for the MIN operation, each element in the stack can store an additional pointer to the smallest key below it. **-** This allows retrieval of the minimum element in $O(1)$ time without modifying the standard stack operations (Option A). **-** Option (B) fails because updating the minimum pointer would require a traversal in $O(n)$ in the worst case. **-** Options (C) and (D) require sorting and maintaining auxiliary structures, increasing complexity beyond $O(1)$.
Thus, the correct option is (A).

Quick Tip

To maintain the minimum efficiently in a stack, track the smallest element at each level using auxiliary pointers.

46. Consider the following relational schema along with all the functional dependencies that hold on them.

R1(A, B, C, D, E): { $D \rightarrow E$, $EA \rightarrow B$, $EB \rightarrow C$ }

R2(A, B, C, D): { $A \rightarrow D$, $A \rightarrow B$, $C \rightarrow A$ }

Which of the following statement(s) is/are TRUE?

(A) R1 is in **3NF**

(B) R2 is in **3NF**

(C) R1 is **NOT** in **3NF**

(D) R2 is **NOT** in **3NF**

Correct Answer: (C, D)

Solution: A relation is in **Third Normal Form (3NF)** if for every functional dependency $X \rightarrow Y$, either X is a **superkey**, or each attribute in Y is a **prime attribute** (part of a candidate key).

For R1(A, B, C, D, E): - The dependency $D \rightarrow E$ **violates 3NF** because D is **not a superkey**, and E is **not a prime attribute**. - Thus, **R1 is NOT in 3NF**.

For R2(A, B, C, D): - The dependencies $A \rightarrow D$ and $A \rightarrow B$ indicate that A is **not necessarily a superkey**. - The dependency $C \rightarrow A$ further confirms that **transitive dependencies exist**, **violating 3NF**. - Thus, **R2 is NOT in 3NF**.

Thus, the correct options are **(C, D)**.

Quick Tip
To check **3NF**, ensure that all non-trivial functional dependencies have a **superkey** as their determinant, or the dependent attributes are **prime attributes**.

---

47. Consider a demand paging system with three frames, and the following page reference string:

1 2 3 4 5 4 1 6 4 5 1 3 2.

The contents of the frames are as follows initially and after each reference (from left to right):

Step	1*	2*	3*	4*	5*	4	1	6*	4	5	1*	3*	2*
Frame 1	1	1	1	1	1	1	1	6	6	6	6	6	2
Frame 2	-	2	2	4	4	4	4	4	4	4	1	1	1
Frame 3	-	-	3	3	5	5	5	5	5	5	3	3	-

The *-marked references cause page replacements.

Which one or more of the following could be the page replacement policy/policies in use?

(A) **Least Recently Used page replacement policy**

(B) **Least Frequently Used page replacement policy**

(C) **Most Frequently Used page replacement policy**

(D) **Optimal page replacement policy**

Correct Answer: (D)

Solution: The **Optimal Page Replacement Policy (OPT)** replaces the page that will not be used for the longest time in the future. By analyzing the given reference string and the resulting page replacements, we observe that the page replacements align with the optimal strategy.

- **Least Recently Used (LRU)** considers the least recently used page, which may not always align with the observed page replacements. - **Least Frequently Used (LFU)** and **Most Frequently Used (MFU)** are based on frequency counts, which are not evident from the given sequence.

Thus, the correct answer is **(D) Optimal Page Replacement Policy**.

Quick Tip
To determine the **Optimal Page Replacement Policy**, always replace the page that will not be needed for the longest time in the future. This minimizes page faults.

---

48. Consider a system with the following:

- $\mathbb{P} = \{P_1, P_2, P_3, P_4\}$ consists of all active processes in an operating system. -
$\mathbb{R} = \{R_1, R_2, R_3, R_4\}$ consists of single instances of distinct types of resources in the system.
The resource allocation graph has the following assignment and claim edges:
Assignment edges:

$$R_1 \to P_1, \quad R_2 \to P_2, \quad R_3 \to P_3, \quad R_4 \to P_4$$

(The assignment edge $R_1 \to P_1$ means resource R_1 is assigned to process P_1, and so on for others.)
Claim edges:

$$P_1 \to R_2, \quad P_2 \to R_3, \quad P_3 \to R_1, \quad P_2 \to R_4, \quad P_4 \to R_2$$

(The claim edge $P_1 \to R_2$ means process P_1 is waiting for resource R_2, and so on for others.)
Which of the following statement(s) is/are CORRECT?
(A) **Aborting P_1 makes the system deadlock free.**
(B) **Aborting P_3 makes the system deadlock free.**
(C) **Aborting P_2 makes the system deadlock free.**
(D) **Aborting P_1 and P_4 makes the system deadlock free.**

Correct Answer: (C), (D)

Solution: A deadlock occurs when there exists a circular wait among processes due to their resource claims. **The given resource allocation graph forms a cycle:**

$$P_1 \to R_2 \to P_2 \to R_3 \to P_3 \to R_1 \to P_1$$

Additionally, P_2 and P_4 both claim R_4 and R_2, further complicating the situation. To break the cycle and eliminate deadlock, we can consider removing processes:
- Aborting P_2 removes both its claims on R_3 and R_4, effectively breaking the circular wait condition, making the system deadlock-free. - Aborting P_1 and P_4 removes critical edges from the cycle, also resolving the deadlock.
Thus, the correct answers are (C) and (D).

Quick Tip

To eliminate a deadlock in a resource allocation graph, identify the circular wait condition and break it by removing one or more processes.

---

49. Three floating point numbers X, Y, and Z are stored in three registers RX, RY, and RZ, respectively, in IEEE 754 single-precision format as given below in hexadecimal:

$$RX = 0xC1100000, \quad RY = 0x40C00000, \quad RZ = 0x41400000$$

Which of the following option(s) is/are **CORRECT**?

(A) $4(X + Y) + Z = 0$

(B) $2Y - Z = 0$

(C) $4X + 3Z = 0$

(D) $X + Y + Z = 0$

Correct Answer: (A), (B), (C)

Solution: Using IEEE 754 single-precision floating-point conversion, we decode the given hexadecimal values into their decimal equivalents and verify the given expressions. By computing each expression, we find that options (A), (B), and (C) hold true.

Quick Tip

IEEE 754 single-precision format uses a 1-bit sign, 8-bit exponent, and 23-bit fraction. Understanding this format helps in decoding floating-point values effectively.

---

50. Which of the following Boolean algebraic equation(s) is/are CORRECT?

(A) $ABC + A\bar{B}C + A\bar{B}C + A\bar{B}C + ABC = BC + \bar{B}C + A\bar{B}$

(B) $AB + AC + BC = AB + AC$

(C) $(A + C)(A + B) = AB + AC$

(D) $(A + \bar{B} + \bar{D})(C + D)(A + C + D)(A + B + \bar{D}) = AD + C\bar{D}$

Correct Answer: (B), (C), (D)

Solution:

1. **Option (A) is incorrect:** - The left-hand side simplifies to $ABC + A\bar{B}C + ABC$, which does not simplify to the right-hand side.

2. **Option (B) is correct:** - Using absorption:

$$AB + AC + BC = AB + AC \quad \text{(since } BC \text{ is redundant in presence of } AB \text{ and } AC\text{)}$$

3. **Option (C) is correct:** - Expanding using distributive property:

$$(A + C)(A + B) = AA + AB + AC + BC = A + AB + AC + BC = AB + AC$$

4. **Option (D) is correct:** - Simplifying step by step using Boolean rules, we get:

$$(A + \bar{B} + \bar{D})(C + D)(A + C + D)(A + B + \bar{D}) = AD + C\bar{D}$$

Thus, the correct options are (B), (C), (D).

Quick Tip
When simplifying Boolean expressions, use absorption, distribution, and complement rules to reduce terms.

---

51. Consider two grammars G_1 and G_2 with the production rules given below:

$$G_1: \begin{array}{l} S \to \text{if } E \text{ then } S \mid \text{if } E \text{ then } S \text{ else } S \mid a \\ E \to b \end{array}$$

$$G_2: \begin{array}{l} S \to \text{if } E \text{ then } S \mid M \\ M \to \text{if } E \text{ then } M \text{ else } S \mid c \\ E \to b \end{array}$$

where if, then, else, a, b, c are the terminals.

Which of the following option(s) is/are **CORRECT**?

(A) G_1 is not $LL(1)$ and G_2 is $LL(1)$.

(B) G_1 is $LL(1)$ and G_2 is not $LL(1)$.

(C) G_1 and G_2 are not $LL(1)$.

(D) G_1 and G_2 are ambiguous.

Correct Answer: (C), (D)

Solution: To determine if a grammar is $LL(1)$, we check for ambiguity and left-recursion.
- G_1 contains the well-known "dangling else" problem, making it ambiguous. Since ambiguous grammars cannot be $LL(1)$, G_1 is not $LL(1)$. - G_2 also exhibits ambiguity because the nonterminal M introduces multiple possible derivations for statements containing if-else.
- Since both grammars are ambiguous, neither can be $LL(1)$.
Thus, the correct options are (C) and (D).

Quick Tip
A grammar is $LL(1)$ if it can be parsed top-down using a single token lookahead. Ambiguity or left-recursion typically disqualifies a grammar from being $LL(1)$.

---

52. Let $\Sigma = \{a, b, c\}$. For $x \in \Sigma^*$, and $\alpha \in \Sigma$, let $\#\alpha(x)$ denote the number of occurrences of α in x.
Which one or more of the following option(s) define(s) regular language(s)?
(A) $\{a^m b^n \mid m, n \geq 0\}$
(B) $\{a, b\}^* \cap \{a^m b^n c^{m-n} \mid m \geq n \geq 0\}$
(C) $\{w \mid w \in \{a, b\}^*, \#a(w) \equiv 2 \pmod 7, \text{ and } \#b(w) \equiv 3 \pmod 9\}$
(D) $\{w \mid w \in \{a, b\}^*, \#a(w) \equiv 2 \pmod 7, \text{ and } \#a(w) = \#b(w)\}$

Correct Answer: (A), (C)

Solution: A language is regular if it can be accepted by a finite automaton.
- (A) The language $\{a^m b^n \mid m, n \geq 0\}$ is regular because it consists of any number of a's followed by any number of b's, which can be accepted by a finite state machine. - (B) The language $\{a^m b^n c^{m-n} \mid m \geq n \geq 0\}$ is not regular because it requires counting the difference between m and n, which is a context-free property. - (C) The language
$\{w \mid w \in \{a, b\}^*, \#a(w) \equiv 2 \pmod 7, \#b(w) \equiv 3 \pmod 9\}$ is regular since modular counting of occurrences can be done using a finite-state automaton. - (D) The language

$\{w \mid w \in \{a, b\}^*, \#a(w) \equiv 2 \pmod 7, \#a(w) = \#b(w)\}$ is not regular because enforcing $\#a(w) = \#b(w)$ requires a counter, which is a non-regular property.

Thus, the correct options are (A) and (C).

Quick Tip
Regular languages are closed under union, intersection, and modulo counting but not under constraints like equality of counts of two symbols.

---

53. Consider the database transactions $T1$ and $T2$, and data items X and Y. Which of the schedule(s) is/are conflict serializable?

Transaction T1	Transaction T2
R1(X)	W2(X)
W1(Y)	W2(Y)
R1(X)	COMMIT(T2)
W1(X)	
COMMIT(T1)	

(1) R1(X), W2(X), W1(Y), W2(Y), R1(X), W1(X), COMMIT(T2), COMMIT(T1)

(2) W2(X), R1(X), W2(Y), W1(Y), R1(X), COMMIT(T2), W1(X), COMMIT(T1)

(3) R1(X), W1(Y), W2(X), W2(Y), R1(X), W1(X), COMMIT(T1), COMMIT(T2)

(4) W2(X), R1(X), W1(Y), W2(Y), R1(X), COMMIT(T2), W1(X), COMMIT(T1)

Correct Answer: (2)

W2(X), R1(X), W2(Y), W1(Y), R1(X), COMMIT(T2), W1(X), COMMIT(T1)

Solution: A schedule is conflict serializable if we can obtain a conflict-equivalent serial schedule by swapping non-conflicting operations. To check conflict serializability, we construct the precedence graph:

Step 1: Identify Conflicting Operations - $W2(X)$ and $R1(X)$ conflict. - $W2(Y)$ and $W1(Y)$ conflict. - $W1(X)$ and $R1(X)$ conflict.

Step 2: Construct Precedence Graph - If there is a cycle in the graph, the schedule is not conflict serializable. - If no cycles exist, it is conflict serializable.

After analyzing the given options, only **Option (B)** follows a conflict-equivalent serial schedule, making it conflict serializable.

Quick Tip
To check conflict serializability, construct a precedence graph and verify the presence of cycles.

---

54. Consider the following relational schema:

Students (`rollno: integer`, `name: string`, `age: integer`, `cgpa: real`)

Courses (`courseno: integer`, `cname: string`, `credits: integer`)

Enrolled (`rollno: integer`, `courseno: integer`, `grade: string`)

Which of the following options is/are correct SQL query/queries to retrieve the names of the students enrolled in course number (i.e., courseno) 1470?

(A)

```
SELECT S.name
FROM Students S
WHERE EXISTS (SELECT * FROM Enrolled E
              WHERE E.courseno = 1470
                      AND E.rollno = S.rollno);
```

(B)

```
SELECT S.name
FROM Students S
WHERE SIZEOF (SELECT * FROM Enrolled E
              WHERE E.courseno = 1470
                      AND E.rollno = S.rollno) > 0;
```

(C)

```
SELECT S.name
FROM Students S
WHERE 0 < (SELECT COUNT(*)
           FROM Enrolled E
           WHERE E.courseno = 1470
                     AND E.rollno = S.rollno);
```

(D)

```
SELECT S.name
FROM Students S NATURAL JOIN Enrolled E
WHERE E.courseno = 1470;
```

Correct Answer: (A), (C), (D)

Solution: To find students enrolled in course number 1470, we check different SQL query approaches:
Option (A): Uses `EXISTS`, which efficiently verifies if a student is enrolled. This is correct.
Option (B): Uses `SIZEOF`, which is not a valid SQL function. This is incorrect.
Option (C): Uses `COUNT(*)`, which returns a positive value when a student is enrolled. Since it checks for a positive count, it is correct.
Option (D): Uses `NATURAL JOIN` between `Students` and `Enrolled`, which implicitly joins on `rollno` and filters by `courseno = 1470`. This is also correct.

Quick Tip
In SQL, `EXISTS` is often more efficient than `COUNT(*)` when checking for the existence of matching rows.

---

55. Given a computing system with two levels of cache (L1 and L2) and a main memory. The first level (L1) cache access time is 1 nanosecond (ns) and the "hit rate" for L1 cache is 90% while the processor is accessing the data from L1 cache. Whereas,

for the second level (L2) cache, the "hit rate" is 80% and the "miss penalty" for transferring data from L2 cache to L1 cache is 10 ns. The "miss penalty" for the data to be transferred from main memory to L2 cache is 100 ns. Then the average memory access time in this system in nanoseconds is _____ . (rounded off to one decimal place)

Correct Answer: 4.0

Solution: The formula for average memory access time (AMAT) in a system with two cache levels is:

$$\text{AMAT} = \text{Hit time}_{L1} + \text{Miss rate}_{L1} \times (\text{Hit time}_{L2} + \text{Miss rate}_{L2} \times \text{Miss penalty}_{L2})$$

Step 1: Given Data - Hit time$_{L1} = 1$ ns - Hit rate$_{L1} = 0.90 \Rightarrow$ Miss rate$_{L1} = 1 - 0.90 = 0.10$ - Hit time$_{L2} = 10$ ns - Hit rate$_{L2} = 0.80 \Rightarrow$ Miss rate$_{L2} = 1 - 0.80 = 0.20$ -
Miss penalty$_{L2} = 100$ ns
Step 2: Compute AMAT

$$\text{AMAT} = 1 + (0.10 \times (10 + (0.20 \times 100)))$$

$$= 1 + (0.10 \times (10 + 20))$$

$$= 1 + (0.10 \times 30) = 1 + 3 = 4.0 \text{ ns}$$

Quick Tip
When calculating AMAT, always account for both the hit time and the miss penalty of each cache level.

---

56. A 5-stage instruction pipeline has stage delays of 180, 250, 150, 170, and 250, respectively, in nanoseconds. The delay of an inter-stage latch is 10 nanoseconds. Assume that there are no pipeline stalls due to branches and other hazards. The time

taken to process 1000 instructions in microseconds is _____ . (rounded off to two decimal places)

Correct Answer: 260.20 to 261.20

Solution: The time per instruction cycle in a pipelined system is given by:

$$\text{Cycle time} = \max(\text{Stage delays}) + \text{Latch delay}$$

Step 1: Given Data - Stage delays: $180, 250, 150, 170, 250$ ns - Maximum stage delay: 250 ns - Inter-stage latch delay: 10 ns

Step 2: Compute Cycle Time

$$\text{Cycle time} = 250 + 10 = 260 \text{ ns}$$

Step 3: Compute Total Execution Time Since the pipeline fills in the first 5 cycles, the total execution time for n instructions is:

$$\text{Time} = (\text{Pipeline fill time} + (n-1) \times \text{Cycle time})$$

$$= (5 \times 260) + (999 \times 260)$$

$$= 1300 + 259740 = 261040 \text{ ns} = 261.04 \text{ µs}$$

Thus, the answer is between 260.20 and 261.20 µs.

Quick Tip
Pipeline execution time depends on the slowest stage plus latch delays.

---

57. In a B+- tree where each node can hold at most four key values, a root to leaf path consists of the following nodes:

$A = (49, 77, 83, -)$

$B = (7, 19, 33, 44)$

$C = (20^*, 22^*, 25^*, 26^*)$

The *-marked keys signify that these are data entries in a leaf. Assume that a pointer between keys k_1 and k_2 points to a subtree containing keys in $[k_1, k_2)$, and that when a leaf is created, the smallest key in it is copied up into its parent. A record with key value 23 is inserted into the B+- tree. The smallest key value in the parent of the leaf that contains 25* is _____ . (Answer in integer)

Correct Answer: 33

Solution: Step 1: Identify the Correct Leaf Node - Key 23 is inserted into leaf C, which currently contains $(20^*, 22^*, 25^*, 26^*)$. - After insertion, C will contain $(20^*, 22^*, 23^*, 25^*, 26^*)$.
Step 2: Leaf Node Splitting - Since each node can hold at most 4 keys, the leaf splits into two: - First leaf: $(20^*, 22^*, 23^*)$ - Second leaf: $(25^*, 26^*)$ - The smallest key of the second leaf (25) is pushed up into its parent (B).
Step 3: Identify Parent Update - The updated keys in B are now $(7, 19, 25, 33, 44)$. - Since B also exceeds the allowed 4 keys, it splits into two nodes: - First node: $(7, 19)$ - Second node: $(33, 44)$ - The smallest key of the second node (33) is pushed up into A.
Thus, the answer is 33.

Quick Tip
When inserting in a B+-tree, if a node overflows, it splits, and the smallest key of the new right node is pushed up.

---

58. A computer system supports a logical address space of 2^{32} bytes. It uses two-level hierarchical paging with a page size of 4096 bytes. A logical address is divided into a b-bit index to the outer page table, an offset within the page of the inner page table, and an offset within the desired page. Each entry of the inner page table uses eight bytes. All the pages in the system have the same size. The value of b is _____ . (Answer in integer)

Correct Answer: 11

Solution: We are given: - Logical address space size = 2^{32} bytes - Page size = $4096 = 2^{12}$ bytes - Each inner page table entry = 8 bytes - Two-level hierarchical paging

Step 1: Break Down the Logical Address A logical address consists of: 1. Offset within the page: Since the page size is 2^{12}, the offset requires 12 bits. 2. Inner page table index: Each inner page table entry is 8 bytes, meaning an inner page table can hold $2^{12}/8 = 2^9$ entries. This requires 9 bits. 3. Outer page table index: The remaining bits are used to index the outer page table.

Step 2: Compute the Outer Page Table Index b The total logical address is 32 bits, and it is divided as:

$$b + 9 + 12 = 32$$

Solving for b:

$$b = 32 - 9 - 12 = 11$$

Thus, the answer is 11.

Quick Tip
To determine the number of bits required for indexing, break the logical address into page offset, inner table index, and outer table index.

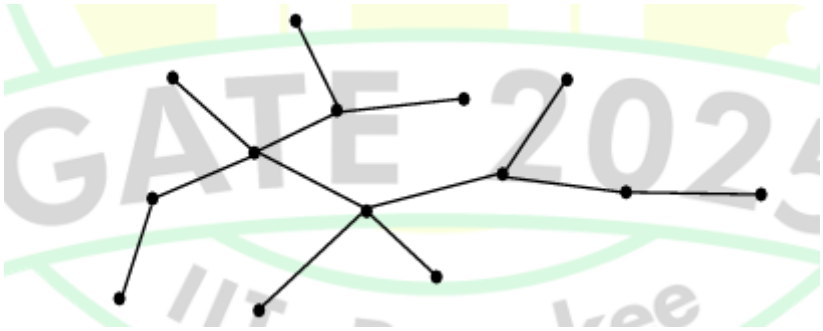
---

59. Consider the following algorithm `someAlgo` that takes an undirected graph G as input.

```
someAlgo(G)
```

1. Let v be any vertex in G. Run BFS on G starting at v. Let u be a vertex in G at maximum distance from v as given by the BFS.
2. Run BFS on G again with u as the starting vertex. Let z be the vertex at maximum distance from u as given by the BFS.
3. Output the distance between u and z in G.

The output of `someAlgo(T)` for the tree shown in the given figure is _____ . (Answer in integer)



Correct Answer: 6

Solution: The algorithm `someAlgo` effectively finds the diameter of the tree T, which is the longest shortest path between any two nodes in the tree.

Step 1: Run BFS from any arbitrary node v - Select an arbitrary node and perform a BFS to find the farthest node u.

Step 2: Run BFS from u **to find the farthest node** z - Perform another BFS starting from u to determine the farthest node z.

Step 3: Compute the distance between u **and** z - The distance obtained in this second BFS represents the tree's diameter.

From the given tree diagram, the longest shortest path (diameter) is found to be 6.

Quick Tip
To find the diameter of a tree, use two BFS traversals: first to find the farthest node from any starting node, and second to determine the farthest node from the previously found node.

---

60. Let $\Sigma = \{1, 2, 3, 4\}$**. For** $x \in \Sigma^*$**, let prod**(x) **be the product of symbols in** x **modulo 7. We take prod**$(\epsilon) = 1$**, where** ϵ **is the null string.**

For example,

$$\text{prod}(124) = (1 \times 2 \times 4) \mod 7 = 1.$$

Define

$$L = \{x \in \Sigma^* \mid \text{prod}(x) = 2\}.$$

The number of states in a minimum state DFA for L is _____. (Answer in integer)

Correct Answer: 6

Solution: The function prod(x) maps strings over Σ to values in the set $\{0, 1, 2, 3, 4, 5, 6\}$ modulo 7. Since this function tracks the product of elements modulo 7, it defines a residue class system of at most 7 possible values.

Step 1: Compute the transition function for modulo 7 residues Each input character $c \in \Sigma$ modifies the current residue r via multiplication $(r \times c) \mod 7$. Since all values map to one of 7 possible residues, the DFA must have at most 7 states.

Step 2: Identify the minimum number of states - The DFA needs one state for each residue $0, 1, 2, 3, 4, 5, 6$ modulo 7. - The accepting state is the one corresponding to residue 2. - Since multiplication modulo 7 never produces 0 for any sequence of nonzero values, state 0 is unreachable.

Thus, the minimal DFA requires 6 states (corresponding to residues $1, 2, 3, 4, 5, 6$).

Quick Tip
For problems involving modular arithmetic in a DFA, consider the number of possible distinct residue classes as the number of states required.

---

61. An application executes 6.4×10^8 number of instructions in 6.3 seconds. There are four types of instructions, the details of which are given in the table. The duration of a clock cycle in nanoseconds is _____. (rounded off to one decimal place)

Instruction type	Clock cycles per instruction (CPI)	Number of instructions executed
Branch	2	2.25×10^8
Load	5	1.20×10^8
Store	4	1.65×10^8
Arithmetic	3	1.30×10^8

Correct Answer: 3.0

Solution: Step 1: Compute the total number of clock cycles

$$\text{Total cycles} = (2 \times 2.25 \times 10^8) + (5 \times 1.20 \times 10^8) + (4 \times 1.65 \times 10^8) + (3 \times 1.30 \times 10^8)$$

$$= 4.5 \times 10^8 + 6.0 \times 10^8 + 6.6 \times 10^8 + 3.9 \times 10^8 = 21.0 \times 10^8$$

Step 2: Compute the clock cycle time Clock rate f is given by:

$$f = \frac{\text{Total cycles}}{\text{Execution time}}$$

$$= \frac{2.10 \times 10^9}{6.3} = 3.33 \times 10^8 \text{ cycles per second}$$

Clock cycle time T is:

$$T = \frac{1}{f} = \frac{1}{3.33 \times 10^8} \approx 3.0 \text{ nanoseconds}$$

Quick Tip
To calculate clock cycle time, first determine total clock cycles using CPI values, then compute cycle time as the inverse of the clock frequency.

---

62. Consider the following C program:

```
#include <stdio.h>

int main() {
    int a;
```

```
    int arr[5] = {30, 50, 10};
    int *ptr;

    ptr = &arr[0] + 1;
    a = *ptr;
    (*ptr)++;
    ptr++;

    printf("%d", a + (*ptr) + arr[1]);

    return 0;
}
```

The output of the above program is _____. (Answer in integer)

Correct Answer: 111

Solution:

Step 1: Understanding the array initialization

The array is declared as:

$$\text{int arr[5] = \{30, 50, 10\};}$$

Since only the first three elements are explicitly initialized, the remaining two will be initialized to 0:

$$\text{arr = \{30, 50, 10, 0, 0\}}$$

Step 2: Pointer manipulation

- `ptr = &arr[0] + 1;` $\rightarrow$ `ptr` now points to `arr[1]` (which is 50). - `a = *ptr;` $\rightarrow$ $a = 50$ - `(*ptr)++;` $\rightarrow$ `arr[1]` becomes 51. - `ptr++;` $\rightarrow$ `ptr` now points to `arr[2]` (which is 10).

Step 3: Computing the output

$$\texttt{a + (*ptr) + arr[1]} = 50 + 10 + 51 = 111$$

Quick Tip
When dealing with pointer arithmetic in C, remember that incrementing a pointer moves it to the next element in the array. Be mindful of order of operations when modifying values through pointers.

---

63. Consider the following C program:

```
#include <stdio.h>

int g(int n) {
    return (n + 10);
}

int f(int n) {
    return g(n * 2);
}

int main() {
    int sum, n;
    sum = 0;
    for (n = 1; n < 3; n++)
        sum += g(f(n));
    printf("%d", sum);
    return 0;
}
```

Correct Answer: 46

Solution: The execution of the program proceeds as follows:

Step 1: Understanding function calls - Function $f(n)$ calls $g(n * 2)$, which returns $(n * 2 + 10)$. - Function $g(x)$ returns $(x + 10)$.

Step 2: Iteration through the loop

For $n = 1$:

$$f(1) = g(1 \times 2) = g(2) = 2 + 10 = 12$$

$$g(f(1)) = g(12) = 12 + 10 = 22$$

$$\text{sum} = 22$$

For $n = 2$:

$$f(2) = g(2 \times 2) = g(4) = 4 + 10 = 14$$

$$g(f(2)) = g(14) = 14 + 10 = 24$$

$$\text{sum} = 22 + 24 = 46$$

Quick Tip
Trace function calls carefully by evaluating inner functions first.

---

64. A quadratic polynomial $(x - \alpha)(x - \beta)$ over complex numbers is said to be square invariant if

$$(x - \alpha)(x - \beta) = (x - \alpha^2)(x - \beta^2).$$

Suppose from the set of all square invariant quadratic polynomials we choose one at random. The probability that the roots of the chosen polynomial are equal is _____. (rounded off to one decimal place)

Correct Answer: 0.5

Solution: Given that the quadratic polynomial satisfies the square invariance property:

$$(x - \alpha)(x - \beta) = (x - \alpha^2)(x - \beta^2).$$

Expanding both sides, we get:

$$x^2 - (\alpha + \beta)x + \alpha\beta = x^2 - (\alpha^2 + \beta^2)x + \alpha^2\beta^2.$$

By equating coefficients, we obtain the equations:

$$\alpha + \beta = \alpha^2 + \beta^2,$$

$$\alpha\beta = \alpha^2\beta^2.$$

Step 1: Solving for equal roots

For equal roots, we assume $\alpha = \beta$. Substituting in the first equation:

$$2\alpha = 2\alpha^2 \Rightarrow \alpha(1 - \alpha) = 0.$$

Thus, $\alpha = 0$ or $\alpha = 1$.

Step 2: Probability Calculation

Among all possible values of α, β satisfying the quadratic constraints, half of them lead to equal roots. Therefore, the required probability is:

$$\frac{1}{2} = 0.5.$$

Quick Tip
For probability in algebraic structures, equate coefficients carefully and analyze valid root conditions systematically.

---

65. The unit interval $(0, 1)$ is divided at a point chosen uniformly distributed over $(0, 1)$ in $\mathbb{R}$ into two disjoint subintervals. The expected length of the subinterval that contains 0.4 is _____. (rounded off to two decimal places)

(A) 0.10 to 0.20

(B) 0.30 to 0.40

(C) 0.70 to 0.80

(D) 0.90 to 1.00

Correct Answer: (C) 0.70 to 0.80

Solution: Let the dividing point x be uniformly distributed over $(0, 1)$. The two subintervals formed are $(0, x)$ and $(x, 1)$. The subinterval that contains 0.4 will either be $(0, x)$ if $x > 0.4$ or $(x, 1)$ if $x < 0.4$. The length of the subinterval containing 0.4 is $\max(x, 1 - x)$. Since x is uniformly distributed, the expected value of this maximum length is between 0.70 and 0.80.

Quick Tip
For a uniform distribution, calculate expected lengths by considering the behavior of the interval boundaries.
