GATE CS & IT 2025 Shift 1 Question Paper with Solutions

Time Allowed :3 Hour	Maximum Marks :100	Total Questions :65

---

1. Given the matrix A, a 3×3 matrix defined as:

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -1 \end{bmatrix}$$

Determine the eigenvalues of A^{13}. **[2]**

Correct Answer:
The eigenvalues of A^{13} are the powers of the eigenvalues of A.
Solution:
Step 1: Calculating Eigenvalues of Matrix A
- To find the eigenvalues of matrix A, we solve its characteristic equation:

$$\det(A - \lambda I) = 0$$

$$A - \lambda I = \begin{bmatrix} 1-\lambda & 1 & 1 \\ 1 & 1-\lambda & 1 \\ 1 & 1 & -1-\lambda \end{bmatrix}$$

- The determinant of this matrix is computed, and we solve for λ. The eigenvalues of matrix A are found to be $\lambda_1 = 3, \lambda_2 = -1, \lambda_3 = -1$.
Step 2: Finding the Eigenvalues of A^{13}
- The eigenvalues of A^{13} are simply the eigenvalues of A raised to the 13th power. Thus, we have:

$$\lambda_1^{13} = 3^{13}, \quad \lambda_2^{13} = (-1)^{13} = -1, \quad \lambda_3^{13} = (-1)^{13} = -1$$

Step 3: Conclusion
- The eigenvalues of A^{13} are $3^{13}, -1, -1$.

> Quick Tip
>
> **Eigenvalues of Matrix Powers:** To find the eigenvalues of A^n, simply raise the eigenvalues of A to the power n.

---

2. Match the following network layers with their respective functions:

- Networking Layer
- Transport Layer

- Data Link Layer

Options for matching:

- Host-to-host communication
- Data packet transfer
- Error detection and correction

Correct Answer:

- Networking Layer: **Data packet transfer**
- Transport Layer: **Host-to-host communication**
- Data Link Layer: **Error detection and correction**

Solution:
Step 1: Understanding Each Layer's Function
- **Networking Layer (Layer 3)**: This layer is responsible for routing data packets across different networks, handling the forwarding of data between devices and managing packet addressing.
- **Transport Layer (Layer 4)**: It ensures reliable communication between two host machines by managing data flow control and error handling.
- **Data Link Layer (Layer 2)**: This layer is responsible for reliable data transmission over the physical layer. It ensures error-free data transfer between adjacent network nodes, typically using error detection and correction techniques.
Step 2: Explanation of the Matches

- **Networking Layer:** It handles data packet forwarding, making "Data packet transfer" the correct match.
- **Transport Layer:** It manages end-to-end communication between hosts, so "Host-to-host communication" is the correct match.
- **Data Link Layer:** It provides error checking and correction during data transfer between devices on the same network, so "Error detection and correction" is the correct function.

Quick Tip

OSI Model Layers:

- Layer 2 (Data Link Layer) is responsible for reliable communication over the physical layer.
- Layer 3 (Networking Layer) focuses on routing and forwarding packets.
- Layer 4 (Transport Layer) ensures reliable communication between hosts.

---

3. What will be the output of the following C code?

```
void foo(int *p, int x) {
    *p = x;
}

void main() {
    int *z;
    int a = 20, b = 25;
    z = &a;
    foo(z, b);
    printf("%d", a);
}
```

Correct Answer: 25
Solution:
Step 1: Code Explanation
- In the 'main' function, the variables $a = 20$ and $b = 25$ are initialized. - A pointer variable z is declared and assigned the address of a using the statement $z = \&a$.
Step 2: Function Call 'foo(z, b)'
- The function 'foo' accepts a pointer to an integer and an integer. Inside the function, the value at the address pointed to by p (which is z) is set to x. - Since z points to a, the function modifies the value of a by setting $*z = 25$.
Step 3: Conclusion
- After calling 'foo', the value of a becomes 25. - The 'printf' statement outputs the value of a, which is 25.
Final Output:

$$\boxed{25}$$

Quick Tip

In C, when a pointer is passed to a function, modifying the value pointed to by the pointer directly affects the original variable in the calling function.

---

4. What is the signed binary expansion of the number -6?

Correct Answer: 11010 (in 2's complement representation for 5-bit binary).
Solution:
Step 1: Understanding Signed Binary Representation
- In signed binary representation, the two's complement form is used for representing negative numbers. The most significant bit (MSB) indicates the sign of the number: '0' represents positive and '1' represents negative.
Step 2: Finding Binary Representation for 6
- Start by finding the binary representation of the positive number 6 in binary. This is 0110_2 for a 4-bit binary representation.
Step 3: Convert to Two's Complement to Get -6

- To get the two's complement representation for -6: 1. Write the binary for +6: 0110_2. 2. Invert all the bits: 1001_2. 3. Add 1: $1001_2 + 1 = 1010_2$.
- So, the two's complement representation of -6 in 4-bit binary is 1010_2.

Step 4: Extending to 5 Bits

- For 5-bit signed expansion, we add an additional bit to maintain the negative sign. This bit will be '1', resulting in 11010_2.

Final Answer:

$$\boxed{11010_2}$$

> Quick Tip
>
> To convert a negative number into its signed binary form using two's complement:
>
> - Start with the binary of the positive number.
> - Invert all bits and add 1 to the result.

---

5. Which of the following values is the greatest?

$$0.\overline{6}, \quad 0.666, \quad \frac{5}{6}, \quad \frac{2}{3}$$

Correct Answer: 0.666

Solution:

Step 1: Convert the decimals to their fractional form

- The repeating decimal $0.\overline{6}$ (i.e., $0.6666\ldots$) is equivalent to the fraction $\frac{2}{3}$. - The decimal 0.666 is a terminating decimal and can be written as $\frac{2}{3}$. - The fraction $\frac{5}{6}$ is already in its simplest form. - The fraction $\frac{2}{3}$ is also in its simplest form.

Step 2: Compare the values

- The repeating decimal $0.\overline{6}$ is equal to $\frac{2}{3}$, which is slightly less than 0.666, as 0.666 is a finite approximation that is just above $\frac{2}{3}$. - 0.666 is the greatest among the options, as it is very close to $\frac{5}{6}$, but still smaller than $\frac{5}{6}$.

Step 3: Conclusion

- Therefore, 0.666 is the greatest value among the listed options.

Final Answer:

$$\boxed{0.666}$$

> Quick Tip
>
> To compare repeating and terminating decimals, first convert them to fractions. Then, compare the fractions to determine the largest value.

---

6. Given the following parameters:

- Logical Address (L.A.): 32 bits
- Physical Address (P.A.): 20 bits

- Page Size (P.S.): 2048 bytes (2 KB)

What is the maximum number of entries in the page table?

- 1. 2^{10}
- 2. 2^{12}
- 3. 2^{20}
- 4. 2^{22}

Correct Answer: 2^{21}
Solution:
Step 1: Address Breakdown
- The logical address space (L.A.) is 32 bits, meaning the total number of bytes addressable is 2^{32} bytes. - The physical address space (P.A.) is 20 bits, which means the physical memory can address 2^{20} bytes.
Step 2: Calculating the Page Size
- The page size is 2048 bytes, or 2^{11} bytes.
Step 3: Number of Pages in the Logical Address Space
- To find the number of pages in the logical address space, divide the total number of bytes by the page size:

$$\frac{2^{32}}{2^{11}} = 2^{21}$$

Step 4: Number of Frames in the Physical Address Space
- The number of frames in the physical address space is:

$$\frac{2^{20}}{2^{11}} = 2^{9}$$

Step 5: Number of Entries in the Page Table
- The page table entries correspond to the number of pages in the logical address space. Therefore, the number of entries in the page table is 2^{21}.
Final Answer:

$$\boxed{2^{21}}$$

Quick Tip

To calculate the number of entries in a page table, divide the total logical address space by the page size to determine the number of pages. This number represents the number of entries in the page table.

---

7. Which of the following correctly represents the 3-way handshake in TCP connection establishment?

- (A) SYN → SYN-ACK → ACK
- (B) SYN → ACK → SYN-ACK

- (C) ACK → SYN → SYN-ACK
- (D) SYN → ACK → ACK

Correct Answer: (A) SYN → SYN-ACK → ACK
Solution:
Step 1: Understanding the 3-Way Handshake
The 3-way handshake is a process that establishes a TCP connection between a client and a server. It involves the following steps:
- **Step 1: SYN (Synchronize)**
The client sends a SYN (synchronize) packet to the server to initiate the connection, along with an initial sequence number.
- **Step 2: SYN-ACK (Synchronize-Acknowledgment)**
The server responds with a SYN-ACK packet. This packet acknowledges the client's SYN request and includes its own SYN message for the client.
- **Step 3: ACK (Acknowledgment)**
The client acknowledges the server's SYN-ACK by sending an ACK (acknowledgment) packet back to the server, completing the connection establishment.
Step 2: Conclusion
Thus, the correct sequence of the 3-way handshake is:

$$\boxed{\text{SYN} \rightarrow \text{SYN-ACK} \rightarrow \text{ACK}}$$

Quick Tip

The 3-way handshake ensures reliable communication between a client and server by establishing the initial connection. It is crucial to ensure the SYN, SYN-ACK, and ACK steps occur in that specific order.

---

8. What is the time complexity of the following algorithm?

```
int func(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            printf("Hello");
        }
    }
}
```

Options:

- 1. $O(n)$
- 2. $O(n^2)$
- 3. $O(n^3)$
- 4. $O(\log n)$

Correct Answer: $O(n^2)$
Solution:
Step 1: Analyzing the Algorithm
- The outer loop runs n times.
- For each iteration of the outer loop, the inner loop also runs n times.
- The total number of executions of the `printf` statement is $n \times n = n^2$.
Step 2: Conclusion
- Therefore, the time complexity of the given algorithm is $O(n^2)$.
Final Answer:

$$\boxed{O(n^2)}$$

Quick Tip

To determine the time complexity of an algorithm, multiply the number of iterations of each nested loop: - If a loop runs n times inside another loop running n times, the time complexity is $O(n^2)$.

---

9. Consider the following sequence of operations on an initially empty stack:

- Push 10
- Push 20
- Pop
- Push 30
- Pop
- Push 40

What will be the final content of the stack?
Options:

- 1. 10, 40
- 2. 40, 20
- 3. 40, 10
- 4. 30, 40

Correct Answer: 10, 40
Solution:
Step 1: Breakdown of the Stack Operations
- Push 10: Stack = [10]
- Push 20: Stack = [10, 20]
- Pop: Removes 20, Stack = [10]
- Push 30: Stack = [10, 30]
- Pop: Removes 30, Stack = [10]

- Push 40: Stack = [10, 40]

Step 2: Conclusion

- The final content of the stack is [10, 40].

Final Answer:

$$\boxed{10, 40}$$

> Quick Tip
>
> A stack follows the Last-In, First-Out (LIFO) principle: - "Push" adds an element to the top of the stack, and "Pop" removes the top element. - Always track the sequence of push and pop operations to determine the final stack contents.

---

10. Which of the following statements about the binary search algorithm is correct when applied to a sorted array?

Options:

- 1. It always has a time complexity of $O(n)$
- 2. It works on unsorted arrays
- 3. It divides the search space into two equal halves on each iteration
- 4. It requires the array to be sorted in descending order

Correct Answer: 3. It divides the search space into two equal halves on each iteration

Solution:

Step 1: Understanding Binary Search

- Binary search is an efficient algorithm used to search for an element in a sorted array. - It works by repeatedly dividing the search space into two equal halves and checking if the element is in the left or right half.

Step 2: Conclusion

- The correct statement is: "It divides the search space into two equal halves on each iteration."

Final Answer:

$$\boxed{\text{3.It divides the search space into two equal halves on each iteration.}}$$

> Quick Tip
>
> Binary search is efficient with $O(\log n)$ time complexity, but it only works on sorted arrays: - The algorithm divides the search space in half each time, ensuring fast searches. - Always ensure the array is sorted before using binary search.