Lecturer Exam Suggestion

1) Object-Oriented Programming (OOP)

a) Class & Object
b) Feature of OOP (Data Abstraction, Encapsulation, Polymorphism, Inheritance)
c) Access Specifiers (public, private, protected)
d) Constructors and Destructors
e) Virtual Functions & Pure Virtual Functions (C++)
f) Function Overloading vs Overriding
g) Friend Functions & Friend Classes (C++)
h) Templates (Function & Class Templates) (C++)
i) Exception Handling (try-catch-finally blocks)
j) Interface and it uses (Java)
k) Code example of Data Abstraction, Encapsulation, Polymorphism, Inheritance in both C++ and Java

2) Algorithm and Data Structure

a) Time and Space Complexity calculation
b) Bubble Sort, Merge Sort, Quick Sort, Heap Sort, Radix Sort algorithm simulation with time and space complex analysis.
c) Searching Algorithms (Linear Search, Binary Search, Ternary Search) code and simulation both.
d) Recursion with code example (Factorial, Fibonacci, Tower of Hanoi).
e) Stack and Queue concept and code.
f) Linked Lists concept and code.
g) Difference between Array and Linked Lists.
h) Greedy Algorithms (Huffman Coding, Kruskal's & Prim's Algorithms, Fractional Knapsack) only concept/simulation not code.
i) Dynamic Programming (Longest Common Subsequence, Bellman–Ford Shortest Path, Floyd-Warshall, and Matrix Chain Multiplication.) only concept/simulation not code.
j) Graph Algorithms (BFS, DFS, Dijkstra, Bellman-Ford, Floyd-Warshall, Kruskal's & Prim's) only concept/simulation not code.
k) Tree (Tree traversal (inorder, preorder, postorder), Binary tree, complete binary tree)
l) B-Tree, B+ Tree (just concepts and Difference)
m) Hashing and Hash table concept

3) Databases

a) Primary, foreign and candidate key concept and difference.
b) Join all concept (Inner, Outer, Natural)
c) Basic operations (select, project, union, etc.)
d) DDL, DML concept and commands.
e) Aggregate functions (COUNT, SUM, GROUP BY, HAVING)
f) Normalization (1NF to BCNF) at least 1NF to 3NF
g) ACID properties
h) Deadlock and its prevention

i)
j) ER (Entity-Relationship) model and EER diagrams
k) SQL (Find second largest value, Find second Lowest value, etc)
l) Indexing concept.
m) SQL Injection

4) Operating Systems

a) Process, Thread concept and difference.
b) Multithreading, Multiprocessing concept and difference.
c) Process Scheduling Algorithms (FCFS, SJF, Round Robin, Priority)
d) Paging and Segmentation
e) Page Replacement Algorithms (First in First Out (FIFO), Optimal Page replacement, Least Recently Used (LRU), Most Recently Used (MRU))

5) Digital Logic Design

a) 1's and 2's complement
b) BCD, Gray Code, Excess-3 Code
c) Why NAND, NOR are Universal Gate.
d) Implement Full adder using Half adder.
e) Subtractor
f) Flip-Flops: SR, JK, D, T (Truth tables, characteristic equations, and Diagram)

6) Computer Network

a) OSI 7-layer model (All layers working process and used protocol)
b) TCP/IP 4-layer model
c) IP address (Class A, B, C, D) range and private IP range
d) Subnet Mask
e) Subnetting & Supernetting
f) Math example (Using both Subnet mask and CIDR notation)
g) TCP vs. UDP
h) Handshaking Theorem

7) Artificial Intelligence

a) Difference between Artificial Intelligence, Machine Learning and Deep Learning.
b) Difference between Supervised, Unsupervised and Self supervised learning.
c) Concept of Reinforcement Learning
d) How can we write research paper like abstract, introduction, methodology and result section?
e) Why we used CNN rather than traditional ML model for Image classification.

Lecturer Exam Suggestion

8) Miscellaneous

a) Software development Life Cycle (SDLC)
b) Software testing
c) MIPS architecture
d) Von Neumann architecture
e) Finite Automata (DFA/NFA)
f) Context-Free Grammars (CFG)
g) MQTT protocal

9) Code

a) Remove all special characters from a string using a single loop and without any built-in library function.
b) Convert an ascending array to descending order using a single loop without using any extra array or built-in library functions.
c) Swap two variables without using a third variable
d) Reverse a string without using library functions
e) Check if a number is prime or print all prime number between 1-100.
f) Fibonacci series with and without recursion.
g) Check a number (not using string) is Palindrome or not.
h) Implement Stack, Queue using array and Linked List.
i) Check a tree is complete binary tree or not.