**DATA STRUCTURE Questions of WBUT Exam**

**Array: ~**

1. Discuss the advantage and disadvantages of linked list over array and write a function to insert an element into a sorted array of descending order. [5] [WBUT-2010]
2. Let the size of the elements stored in 8\*3 matrix be 4 bytes each. If the base address of the matrix is 3500, then find the address of A [4, 2] for both row major & column major cases. What is Sparse Matrix? [5] [WBUT-2007]
3. Each element of an array DATA [20][50] requires 4 bytes of storage. Base address of DATA is 2000, determine the location of DATA [10][10] when the array is stored as row major, column major. What is Sparse Matrix? [4+1] [WBUT-2014]

**Sorting: ~**

1. Short notes: Merge Sort [5] [WBUT-2008, 2010, 2014]
2. Short notes: Radix Sort [5] [WBUT-2014]
3. Write down the algorithm of Merge Sort and explain with example. [5] [WBUT-2007, 2009]
4. Show the operation of Merge Sort with a suitable sample data. [2] [WBUT-2009]
5. Show that the running time for Merge Sort algorithm is O (n log2n). [3] [WBUT-2009]
6. Compare the complexity of Selection sort & Insertion sort. [3] [WBUT-2007]
7. Derive values related to the average case worst case behavior f Bubble Sort algorithm. Also confirm that the best case behavior is O (n). [2] [WBUT-2007]
8. Explain with a suitable example, the principle operation of Quick Sort. Find the complexity of Quick sort algorithm. [5+2/3] [WBUT-2007, 2014]
9. Short notes: Quick Sort [5] [WBUT-2008, 2011]
10. Show that worst case complexity of Quick Sort is O (n2). [5] [WBUT-2009]
11. Short notes: Heap Sort [5] [WBUT-2008]
12. Discuss the timing analysis of Heap sort algorithm. [5] [WBUT-2009]
13. Explain with a suitable example the principles of operation of Heap Sort. [5/4] [WBUT-2006, 2014]
14. Find the time complexity of the above algorithm. [3] [WBUT-2006]
15. Define Heap. [2] [WBUT-2009]

**Searching: ~**

1. Write a recursive algorithm for Binary Search. [4] [WBUT-2008, 2011]
2. Compute the time complexity of your algorithm. [3] [WBUT-2008, 2011]
3. Write down the algorithm of Binary Search and calculate the complexity for best, worst and average cases. [5] [WBUT-2010]
4. What is the advantage of Binary Search over Linear Search? [3] [WBUT-2006]

**Recursion: ~**

1. What is Tail Recursion? Explain with example. [5] [WBUT-2006, 2008, 2010, 2011]
2. Short notes: Recursion vs. Iteration [5] [WBUT-2008]
3. Write a recursive algorithm to print Fibonacci numbers. [5] [WBUT-2009]
4. What do you mean by recursion? Write a C code to implement Tower of Hanoi problem using recursion. [5] [WBUT-2011]
5. Write the recursive function for the Tower of Hanoi problem. Also draw the recursion tree for any set of initial values. [3+2] [WBUT-2007]
6. Write an algorithm to solve the Tower of Hanoi problem. Also calculate the time complexity of your algorithm. [5] [WBUT-2007]
7. Short Notes: Tower of Hanoi Problem and Implementation. [5] [WBUT-2014]
8. Let a & b denote positive integers. Suppose a function Q is defined as follows:
9. Return 0 if a < b
10. Q (a, b) = {Q (a-b, b) + 1 if b < a} Find the value of Q (2, 3) and Q (14, 3) [3] [WBUT-2006]

**Stack: ~**

1. What is Stack? Explain the operations and application of stack. [5] [WBUT-2009, 2011]
2. Write an algorithm to convert an Infix expression into its corresponding Postfix expression using Stack. [5 / 6] [WBUT-2008, 2009, 2011]
3. Convert the following Infix expression into equivalent Postfix expression: [5] [WBUT-2010]

A + B \* C + (D \* E + F) \* G

1. Convert the following Infix expression into equivalent Postfix expression: [5] [WBUT-2006]

A \* (B + D) / E - F \*(G + H / K)

1. Convert the following Infix expression into equivalent Postfix expression: [5] [WBUT-2014]

A \* (B + C ? D) – E ? F \*(G / H)

1. Write an algorithm to evaluate Postfix expression using Stack and hence evaluate the following Postfix expression: 5 + 6 7 + - [5] [WBUT-2010]
2. Define Big ‘O’ notation. What is Stack? Why this is called LIFO? [5] [WBUT-2010]
3. Write short notes: Warshall’s Algorithm [5] [WBUT-2006]
4. What is an Abstract Data Type? Define the ADT for stack. [2+3] [WBUT-2014]
5. Define BIG-O, Ω, θ notation. [2] [WBUT-2014]

**Queue: ~**

1. Compare and contrast between Dequeue and Priority queue. [4] [WBUT-2008]
2. Define Circular Queue. Write an algorithm to insert item in a circular queue. What is input restricted Dequeue / Priority Queue? [2 + 5 +2] [WBUT-2009, 2011, 2014]
3. Why queue is called FIFO? [2] [WBUT-2010]
4. Compare and contrast between De-queue and Priority queue. [4] [WBUT-2011]
5. Short notes: Double ended queue. [5] [WBUT-2011]
6. Short notes: De-Queue – operation & application. [5] [WBUT-2006]
7. Construct the following queue of characters where queue is a circular array which is allocated six memory cells. FRONT=2, REAR=4 & QUEUE: \_, A, C, D, \_, \_. Describe the queue as the following operations take place: [6] [WBUT-2006, 2010]
   1. F is added to the queue.
   2. Two characters are deleted from the queue.
   3. K, L, M are added to the queue.
   4. Two characters are deleted from the queue.
   5. R is added to the queue.
   6. One character is deleted from the queue.

**Linked List: ~**

1. What are the advantage and disadvantages of linked list over array? [5/3] [WBUT-2006, 2008, 2013]
2. What is Doubly Linked List? What are the advantage and disadvantages of Doubly Linked List over Single Linked List? [5] [WBUT-2008]
3. Write an algorithm to reverse a Single Linked List. [3/4/5] [WBUT-2006, 2008, 2010, 2011]
4. Write algorithms to insert and delete elements from Doubly Linked List. [5] [WBUT-2009, 2011, 2014]
5. Write a function to insert data in a Linked List in a sorted manner. [5] [WBUT-2009]
6. What is a Dummy node in a Linked List? [2] [WBUT-2010]
7. Write the function to find the predecessor of a node in a Linked List. [5] [WBUT-2010]
8. Write an algorithm to add two polynomials. [4] [WBUT-2007]
9. How can a polynomial such as 5x8 + 600x5 + 45x2 – 5x + 56 be represented by Linked List?

[5] [WBUT-2010]

1. How can a polynomial such as 6x6 + 4x3 – 2x + 10 be represented by Linked List?

[2] [WBUT-2006]

**Tree: ~**

1. Define and distinguish between Binary Tree and B-Tree. [5] [WBUT-2008, 2011]
2. Prove that for any non empty binary tree T, if N0 is the number of leaves and N2 is the number of nodes of degree 2, then N0 = N2 + 1. [4 / 5] [WBUT-2006, 2007, 2008, 2009, 2010, 2011]
3. Count the total number of nodes of a binary tree having depth n. [1] [WBUT-2006]
4. Write down the non recursive function for in-order traversal of a threaded binary tree. [4] [WBUT-2006, 2008, 2010]
5. Given below the In-order and Pre-order traversal of a Binary Tree. Draw the tree and find the Post-order traversal. [5] [WBUT-2009]

In-order: D I B E J A F C K G Pre-order: A B D I E J C F G K

1. Given below the In-order and Pre-order traversal of a Binary Tree. Draw the tree and state the algorithm to construct tree. Also draw its Post-order Traversal. [5/6/7] [WBUT-2006, 2010, 2014]

In-order: D G B A H E I C F Pre-order: A B D G C E H I F

1. In a non empty Binary tree the following list occurs after tree traversal. Generate the complete Binary Tree indicating all intermediate steps. Hence find Pre-order traversal. [7] [WBUT-2011]

In-order: D G B A H E I C F Post-order: G D B H I E F C A

1. Write an algorithm of In-order traversal of binary tree. [5] [WBUT-2009]
2. Write a C language function to find the recursive function for in-order traversal. [5] [WBUT-2014]
3. What is BST? [2] [WBUT-2006]
4. Write an algorithm to insert a node in a BST. [7] [WBUT-2007]
5. Define Binary Search Tree. Construct a binary search tree with the following elements:

45, 67, 32, 88, 65, 21, 43, 40, 92, 81 [6] [WBUT-2009]

1. Define AVL tree with example. [2] [WBUT-2008, 2010]
2. Construct an AVL tree with the following elements. Clearly mention each rotation.

50, 40, 35, 58, 48, 42, 60, 30, 33, 25 [6] [WBUT-2008]

1. Construct an AVL tree with the following elements. Clearly mention each rotation.

51, 40, 34, 59, 46, 42, 62, 30, 33, 26 [5] [WBUT-2009]

1. Construct an AVL tree with the following elements. Clearly mention each rotation.

g, h, s, l, e, m, t, u [5/6] [WBUT-2006, 2010, 2014]

1. Short notes: AVL Tree [5] [WBUT- 2011]
2. Short notes: Threaded Binary Tree [3/5] [WBUT-2008, 2011, 2014]
3. Short notes: B-Tree [5] [WBUT-2010]
4. Construct a B-Tree of order 5 with the following elements: [5] [WBUT-2009]

11, 72, 61, 20, 110, 40, 80, 130, 100, 42, 191, 92, 181, 242, 32, 122, 140, 162

1. Show the stages in growth of an order-4 B-Tree when the following keys are inserted in the order given: 84, 82, 29, 99, 65, 12, 50, 28, 58, 71, 92, 75 [5] [WBUT-2014]
2. What is threading? [2] [WBUT-2009]
3. What is two-way threading? [3/2] [WBUT-2006, 2010]

**Graph: ~**

1. Define a directed graph. Provide an example. [2+2] [WBUT-2006]
2. Discuss about the following terminology: In-degree, Sink, Cycle, Network. [4] [WBUT-2006]
3. Define adjacency matrix corresponding to a digraph. [2] [WBUT-2006]
4. Draw the graph corresponding to the following bit matrix: [2] [WBUT-2006]
5. What is Path Matrix? [2] [WBUT-2006]
6. What is complete graph? Show that sum of degree of all vertices in a graph is always even.

[5] [WBUT-2008, 2011]

1. Write down BFS algorithm for searching a graph. [5] [WBUT-2008, 2009, 2011]
2. Compare BFS algorithm with DFS. [3] [WBUT-2011]
3. Explain the operation of BFS for the following (Figure – A) undirected graph as follows taking B as source vertex: [5/4] [WBUT-2008, 2011]

V1

V2

V3

V4

V5

V6

1. For the above Graph (Figure – B) Find BFS Traversal and DFS Traversal. [5] [WBUT-2006]

Figure – B

Figure – A

Figure – C

1. Represent the graph (Figure – C) using Adjacency Matrix, Adjacency List. Starting from the vertex V1 show BFS and DFS traversal of the graph. [4+4] [WBUT-2014]

**Hashing: ~**

1. Define Hashing. Explain with suitable example the collision resolution scheme using linear probing with open addressing. [1 + 5] [WBUT-2009, 2010, 2011]
2. What is Hashing? Explain Linear Probing & Quadratic Probing with example. [1+3] [WBUT-2007]
3. Define Hashing. Briefly explain the different commonly used hash functions. Explain different methods of collision resolution techniques. [2+3+5] [WBUT-2014]