ITPC-45 (Advanced Data Structures and Algorithms Lab)

Part-1

1. Write a program to implement Linear Search and Binary Search. Also perform its complexity analysis.
2. Finding kth minimum and maximum element in Heap
3. Implement Quick Sort with duplicate numbers in the given array/elements
4. Build Min heap, Max heap and sort the given elements
5. Read the marks obtained by students of third year in an online examination of particular subject. Find out maximum and minimum marks obtained in that subject. Use heap data structure. Analyze the algorithm
6. Delete kth indexed element in Min heap and Max heap.

Part-2

7. Write a program to implement Merge sort.
8. Write a program to implement Selection sort.
9. Write a program to implement Insertion sort. Compare the complexities of various sorting algorithms.
10. Implement the ascending and descending order using Quick Sort.
11. Implement Quick Sort using first/last/any random element as pivot

Part-3

12. Beginning with an empty binary search tree with complexity analysis, Construct binary search tree by inserting the values in the order given. After constructing a binary tree –
    i. Insert new node
    ii. Find number of nodes in longest path
    iii. Minimum data value found in the tree
    iv. Change a tree so that the roles of the left and right pointers are swapped at every node
    v. Search a value
13. Given sequence k = k1 < … < kn of n sorted keys, with a search probability pi for each key ki . Build the Binary search tree that has the least search cost given the access probability for each key.
14. Write a program to implement AVL Tree with various operations of Insertion, deletion and searching.
15. Write a program to implement RED-BLACK Tree and its operations.
16. Write a program to implement binomial heap and its operations
    I. Creation
    II. Finding the min-heap from binomial heap
    III. Union of two binomial heaps

17. Write a program to implement Fibonacci heap and its operations
    i. Creation
    ii. Inserting a node in binomial heap
    iii. Extracting a minimum node
    iv. Deletion

Part-4

18. Write a program to implement naïve search for pattern matching.
19. Write a program to implement Rabin Karp algorithm for pattern matching.
20. Write a program to implement Knuth Morris Pratt (KMP) algorithm for pattern matching. Compare it with naïve approach.

Part-5

21. Write a program to implement approximate algorithm for vertex cover problem.
22. Write a program to implement approximate algorithm for subset sum problem.
23. Write a program to implement matrix multiplication.