International Institute of Information Technology, Bangalore
EG102 Data Structures and Algorithms Lab.

1. Linked Lists in C : Implement Search Insert, Delete and reverse of a linked list.

2. Stacks and Queues : Implement Stacks and queues, using both arrays and linked list. Implement the algorithms for stack permutation, Min stack, queue using two stacks, histogram, smallest number on the right of an array etc.

3. Let $F(0) = 0, F(1) = 1$ and $F(n) = (F(n-1) + F(n-2))\%m$. Implement the algorithm to compute $F(n)$, by finding the period of the sequence. $1 < n < 10^{10^5}, 10 < m < 10^6$.

4. Let $F(0) = 0, F(1) = 1, F(2) = 2$ and $F(n) = (F(n-1) + F(n-3))\%m$. Implement the algorithm to compute $F(n)$, using the matrix multiplication method . $1 < n < 10^{10^5}, 10 < m < 10^6$.

5. Implement MergeSort, QuickSort and FindRank algorithms.

6. Given an array of integers, $a_0, a_1, a_2, \ldots a_{n-1}$, we would like to know, if there are three numbers such that sum of two numbers is equal to the third number. That is if there is i, j, k such that $a_i + a_j = a_k$. Give an $O(n^2)$ algorithm to solve this problem.

7. Let $a_1, a_2, \ldots a_n$ be a sequence of distinct numbers. The pair (i, j) is called a inversion, if $i < j$ and $a_i > a_j$. Write an $O(n \log n)$ algorithm to determine the number of inversions in the given array.

8. You are given a sequence of $10 < n < 10^9$ numbers and a number $10 < k < 10^5$ such that, $1 \leq k \leq n$. Write an efficient algorithm to list the k smallest numbers among the given sequence of numbers. What is the complexity of the algorithm ?

9. Hashing :

   (a) Implement Robin Karp's string matching algorithm.

   (b) Given two strings, Implement an efficient algorithm using hashing to find the longest repeating substring of the given two sequences.

(c) Implement an efficient algorithm using hashing to find the longest palindrome substring of a given sequence.

10. Implement Pre-Order, Post-order, In- order, Level Order traversal of a tree. Write program to compute the height and level of each node in a binary tree. Given Pre-Order and In- order write a program to build the binary tree.

11. Implement Binary Heap, Binary Index Tree and the Segment Tree.

12. Implement the AVL Tree supporting Insert, Delete, Search, FindRank and Rank operations.

13. Implement DFS, BFS, Dijkstra, Prims and Krushkal's algorithms

14. Implement the algorithm for Topological sort and the strongly connected components of a directed graph.

15. Implement an algorithm to decide if a given graph is a bipartite graph and two find an Euler tour if one exists.