### ****1. Basics of Recursion****

1. What is sorting? Define sorting.
2. Differentiate between internal sorting and external sorting. Why is external sorting slower than internal sorting? Why do you need external sorting?
3. What do you mean by adaptive and non-adaptive sorting?
4. Define stability and efficiency of sorting algorithms.
5. What are the conflicting efficiency considerations in various sorting methods?
6. How can you compare either two sorting or two searching algorithms?
7. Define Big 'O' notation. Compare linear logarithmic, linear, and quadratic order function.
8. Provide the best case, average case, and worst case for the following algorithms in Big-Oh: bubble sort, insertion sort, merge sort, and selection sort.

### ****2. Internal vs External Sorting****

1. What is internal sorting?
2. Define internal and external sorting.
3. What is internal and external sorting? Write an algorithm for shell sort.

### ****3. Comparison of Sorting Algorithms****

1. Compare and contrast the efficiency of Bubble sort, Quick sort, Insertion sort and Selection sort algorithms with an example.
2. Explain which elementary sorting algorithm (i.e., Bubble, Insertion, Selection) you choose when the input data is in almost sorted form.

### ****4. Quick Sort****

1. What is the worst-case time complexity of quicksort?
2. Sort the following data using the quicksort algorithm: 20, 50, 45, 36, 8, 90, 85, 34.
3. Write a recursive algorithm for quicksort and trace the algorithm for the following data: 35, 22, 10, 3, 48, 29, 6, 42, 8, 75.
4. Write an algorithm for quick sort and trace your algorithm for a given sequence of data 5, 43, 99, 20, 45, 7, 6, 63, 92, 4.
5. Write the algorithm of the quick sort including the steps of partition. Discuss the complexity of this algorithm.

### Explain quick sort algorithm with Big-oh notation in best case, average case, and worst case and trace it to sort the data: 8, 10, 5, 12, 14, 5, 7, 13.

### ****5. Merge Sort****

1. Trace the steps to sort the following set of data using merge sort: 85, 76, 46, 92, 30, 41, 12.
2. Discuss merge sort. How do you rate this sorting compared to selection sort?
3. Write an algorithm and C function for merge sort.
4. Trace Merge sort for the following data: 11, 45, 61, 33, 55, 9, 83, 25.

### ****6. Heap Sort****

1. Sort the following elements using a max heap and also analyze its computational complexity: 42, 36, 56, 27, 63, 72, 62, 15.
2. Consider the following max heap: 50, 30, 20, 15, 10, 8, 16. Insert a new node with value 60.
3. Construct a heap sort for the following given list with an algorithm: 37, 33, 26, 92, 57, 18, 48, 25, 12, 86, 42, 22.
4. Create the heap structure from the following sequence data and sort them using heap sort: 12, 10, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, 2.
5. What is a heap?

### ****7. Selection Sort****

1. What is insertion sort? Perform insertion sort on the following data: 14, 33, 27, 10, 35, 19, 42, 44.
2. Sort the following data using the insertion sort algorithm in ascending order: 14, 33, 27, 10, 35, 19, 42, 44.
3. What is insertion sort? Trace and sort the following data using insertion sort: 90, 57, 80, 10, 22, 21, 45, 9, 78.
4. Hand test the insertion sort algorithm with the following array of numbers: 16, 7, 31, 2, 9, 41, -10

### ****8.**** Insertion ****Sort****

1. What is insertion sort? Perform insertion sort on the following data: 14, 33, 27, 10, 35, 19, 42, 44.
2. Sort the following data using the insertion sort algorithm in ascending order: 14, 33, 27, 10, 35, 19, 42, 44.
3. What is insertion sort? Trace and sort the following data using insertion sort: 90, 57, 80, 10, 22, 21, 45, 9, 78.
4. Hand test the insertion sort algorithm with the following array of numbers: 16, 7, 31, 2, 9, 41, -10.

### ****9.**** Shell ****Sort****

### Explain shell sort. Sort the numbers 92, 83, 22, 49, 36, 98, 12, 9, 70, 51 using shell sort.

### ****10.**** Radix ****Sort****

1. Sort the following data using radix sort.
2. Define a radix sort with its algorithm.
3. Explain the basic principle of radix sort. Trace the sorting steps in quick sort and radix sort. 25, 57, 48, 37, 12, 92, 86, 33.

### ****11.**** Bubble ****Sort****

 Explain the Bubble sort algorithm. Illustrate it with an example.

 Write a program in C for bubble sorting.

### ****12.**** Binary Search

 Define binary search with an example.

 Explain a binary search with an example.

 Compare sequential search with binary search.

 What is binary search? Write an algorithm to search an item using binary search.

 Explain the binary search algorithm. Illustrate it with an example.

 Trace Binary Search algorithm for the data: 21, 36, 56, 79, 101, 123, 142, 203 and Search for the values 123 and 153.

 Compare and contrast binary search and binary tree search.

 Discuss the binary search technique along with its efficiency.

### ****13.**** Hashing

1. Consider a hash table of size 10. Using linear probing, insert the keys 62, 37, 36, 44, 67, 91, 82, and 107.

### ****14.**** General Tracing Exercise

1. Trace these sort algorithms for the following data: 11, 45, 61, 33, 55, 9, 83, 25.