Sorted Practical Questions by Topic

Stack (27)

1. Sort a string using stack

2. Reverse a string using stack

3. Implement stack using linked list

4. Check palindrome using stack

5. Reverse a stack using recursion

6. Delete specific node from stack

7. Sort a stack using temporary stack

8. Implement stack using queue

9. Check if a given string is a palindrome using a stack

10. Write the code to check that given input is Balance parenthesis by using stack (Leetcode 20)

11. Valid parentheses (LC 20 modified), get the count of invalid pairs

12. Reverse the string to its own place using stack

13. Convert stack into Queue

14. Stack list to stack linked

15. Delete a middle element of the stack

16. Implement a stack with methods to push, pop and get the current highest number in O(1) complexity

17. Minstack and reverse stack

18. Stack that rejects duplicate values

19. Reverse a stack

20. Convert stack into Queue

21. Program to check if a given string is a palindrome using a stack

22. Valid parenthesis (q:no - 20)

23. Stack using linked list

24. Push pop display

25. Reverse the string to their place using stack

26. Sort a stack

27. Find the occurrence of the character-using using hash table (stack-based context)

Queue (14)

1. Reverse a queue
2. Implement queue using stack
3. Stack and queue conversion
4. Queue list to stack queue
5. Circular queue implementation
6. Implement double-ended queue using linked list
7. Circular queue implementation with max length
8. Reverse a queue
9. Convert stack into Queue
10. Circular queue implementation
11. Enqueue, dequeue, display
12. Circular queue printing - refer more
13. Queue implementation
14. Reverse a string using queue

Hash Table (26)

1. Rehashing implementation
2. Hash table implementation
3. Double hashing
4. Remove duplicates using hash table
5. Find the occurrence of the character-using using hash table
6. Implement a hash table and use it to count the frequency of characters in the string "Mississippi"
7. Using a hash table, implement a function to find the first non-repeating character in "swiss"
8. Hash table collision handling implementation
9. Find frequency of characters in a string using hash table
10. Remove duplicate from a string using hashtable
11. Find the uncommon elements from two different arrays using hash table
12. Using a hash table, implement a function to find the first non-repeating character

13. Find the first non-repeating character from a string using inbuilt hash table (map)

14. Find the occurrence of each character in a string using inbuilt hash table (map)

15. Valid anagram - compare two dictionaries

16. Two sum

17. Hash table implementation with collision handling

18. Collision handling in hashmap implementation code

19. Find frequency of character in string using hash table

20. Find the uncommon elements from two different array using hash table

21. Using a hash table, implement a function to find the first non-repeating character

22. Hash table to check string contains duplicate

23. Using hash table find least occurred number

24. Hash table to find two numbers in an array that add up to a target sum

25. Two sum with hashmap make to O(n) (q:no - 1)

26. Remove duplicates from string using hashtable

Sorting (22)

1. Sort a string using merge sort

2. Implement insertion sort

3. Merge two sorted lists - use merge sort property

4. Sort array of students using merge sort

5. Merge sort implementation

6. Perform merge sort on array of strings

7. Combine 2 sorted arrays into a single sorted array in O(n) time

8. Quick sort without additional arrays

9. Merge two sorted list (q:no - 21)

10. Sort an array of objects based on .amount property

11. Bubble sort

12. Insertion sort

13. Selection sort

14. Quick sort

15. Merge and sort two sorted arrays using merge sort concept

16. Sort an array using quicksort (partially completed)

17. Merge sort

18. Selection Sort

19. Merge 2 sorted array in O(n) time complexity

20. Check if array is sorted or not with linear time complexity

21. Heap sort

22. Sort array of students based on age

Other Algorithms (10)

1. Write a code to find the first missing number in the array

2. Find the subarray with the maximum sum (can use Kadane's algorithm)

3. Find the first missing number from an array of numbers

4. Convert APPLE into - A-pp-ppp-llll-eeeee

5. Binary search using recursion - optimize code

6. Palindrome code - without using string methods

7. Reverse the array and the elements of the array

8. Find the occurrence of each character in a string

9. Find the longest consecutive repeating characters in string

10. Find the second longest word in sentence