QSpiders|JSpiders|PySpiders, NOIDA

B-4, Block-B, Sector-3, Noida

LinkedList Programming Questions by Shambhu Sir

Monday, 11 August 2025

SHAMBHU KUMAR QSpiders |JSpiders | PySpiders, NOIDA

Q1

Delete Node in a Linked List

LinkedList based Programming

Q:2

Design a method to Add node at the end of LinkedList.

LinkedList based Programming

Q:3

Design a method to Print all node values.

LinkedList based Programming

Q:4

Design a method to Add node at the beginning of LinkedList.

LinkedList based Programming

Q:5

Design a method to Count nodes in the LinkedList.

LinkedList based Programming

Q:6

Design a method to get first node values.

LinkedList based Programming

Q:7

Design a method to get last node values.

LinkedList based Programming

Q:8

Get node value at a specific position.

LinkedList based Programming

Q:9

Get node value at middle position.

LinkedList based Programming

Q:10

Design a method to Check if LinkedList is empty.

LinkedList based Programming

Q:11

Design a method to clear the linked list.

LinkedList based Programming

Q:12

Design a method to remove first node.

LinkedList based Programming

Q:13

Design a method to remove last node.

LinkedList based Programming

Q:14

Design a method to remove middle node.

LinkedList based Programming

Q:15

Design a method to Remove node at a specific index .

LinkedList based Programming

Q:16

Design a method to Remove node with a specific value.

LinkedList based Programming

Q:17

Design a method to Remove All node with a specific value.

LinkedList based Programming

Q:18

Design Linked List

LinkedList based Programming

Q:19

Design a method to Search for a value in the LinkedList.

LinkedList based Programming

Q:20

Insert node at a given position in LinkedList.

LinkedList based Programming

Q:21

Iterate through LinkedList using loop.

LinkedList based Programming

Q:22

Iterate using recursion.

LinkedList based Programming

Q:23

Convert LinkedList to array.

LinkedList based Programming

Q:24

Find length using recursion.

LinkedList based Programming

Q:25

Check if LinkedList contains a loop (cycle detection).

LinkedList based Programming

Q:26

Linked List Cycle

LinkedList based Programming

Q:27

Reverse a LinkedList (iterative).

LinkedList based Programming

Q:28

Reverse Linked List

LinkedList based Programming

Q:29

Reverse a LinkedList (recursive).

LinkedList based Programming

Q:30

Reverse a portion of LinkedList (between positions m and n).

LinkedList based Programming

Q:31

Reverse nodes in pairs.

LinkedList based Programming

Q:32

Reverse every k nodes in LinkedList.

LinkedList based Programming

Q:33

Find middle node (using slow-fast pointer).

LinkedList based Programming

Q:34

Middle of the Linked List

LinkedList based Programming

Q:35

Delete the Middle Node of a Linked List

LinkedList based Programming

Q:36

Get nth node from end.

LinkedList based Programming

Q:37

Delete nth node from end.

Remove Nth Node From End of List

LinkedList based Programming

Q:38

Swap two nodes without swapping data.

LinkedList based Programming

Q:39

Swap nodes in pairs

LinkedList based Programming

Q:40

Remove duplicates from a sorted LinkedList.

Remove Duplicates from Sorted List

LinkedList based Programming

Q:41

Remove duplicates from an unsorted LinkedList.

LinkedList based Programming

Q:42

Sort LinkedList (Merge Sort).

LinkedList based Programming

Q:43

Merge two sorted LinkedLists.

Merge Two Sorted Lists

LinkedList based Programming

Q:44

Merge K sorted LinkedLists.

Merge k Sorted Lists

LinkedList based Programming

Q:45

Split LinkedList into two halves.

LinkedList based Programming

Q:46

Check if LinkedList is palindrome.

LinkedList based Programming

Q:47

Rearrange LinkedList in zig-zag fashion.

LinkedList based Programming

Q:48

Detect cycle using Floyd's algorithm

LinkedList based Programming

Q:49

Find the start of the loop in LinkedList

LinkedList based Programming

Q:50

Find length of the loop

LinkedList based Programming

Q:51

Remove loop in LinkedList.

LinkedList based Programming

Q:52

Flatten a multilevel linked list

LinkedList based Programming

Q:53

Clone a LinkedList with random pointers

LinkedList based Programming

Q:54

Add two numbers represented by LinkedLists

Add Two Numbers

LinkedList based Programming

Q:55

Intersection point of two LinkedLists

LinkedList based Programming

Q:56

Check if two LinkedLists are identical

LinkedList based Programming

Q:57

Check if LinkedList is circular

LinkedList based Programming

Q:58

Rotate LinkedList by k places.

LinkedList based Programming

Q:59

Segregate even and odd nodes

DoublyLinkedList based Programming

Q:60

Create a Doubly Linked List and insert 5 elements

DoublyLinkedList based Programming

Q:61

Traverse the list forward and backward

DoublyLinkedList based Programming

Q:62

Count the number of nodes in DLL

DoublyLinkedList based Programming

Q:63

Insert a node at the beginning

DoublyLinkedList based Programming

Q:64

Insert a node at the end.

DoublyLinkedList based Programming

Q:65

Insert a node at the given position.

DoublyLinkedList based Programming

Q:66

Delete the first node.

DoublyLinkedList based Programming

Q:67

Delete the last node.

DoublyLinkedList based Programming

Q:68

Delete the node from a specific position.

DoublyLinkedList based Programming

Q:69

Delete a node by value.

DoublyLinkedList based Programming

Q:70

Search a value in DLL.

DoublyLinkedList based Programming

Q:71

Check if DLL is empty.

DoublyLinkedList based Programming

Q:72

Reverse a Doubly Linked List

DoublyLinkedList based Programming

Q:73

Insert node in a sorted Doubly Linked List

DoublyLinkedList based Programming

Q:74

Convert DLL to a circular DLL and traverse it