Linked List Questions for FAANG Interviews

Basic & Fundamental (Easy-Medium)

These cover foundational patterns like traversal, deletion, and simple manipulation.

- 206. Reverse Linked List
- 21. Merge Two Sorted Lists
- 83. Remove Duplicates from Sorted List
- 203. Remove Linked List Elements
- 876. Middle of the Linked List
- 237. Delete Node in a Linked List
- 234. Palindrome Linked List
- 1290. Convert Binary Number in a Linked List to Integer

Two Pointer Pattern (Cycle, Intersection, etc.)

These are classic FAANG questions testing fast & slow pointer logic.

- 141. Linked List Cycle
- 142. Linked List Cycle II
- 160. Intersection of Two Linked Lists
- 19. Remove Nth Node From End of List
- 143. Reorder List
- 86. Partition List
- 328. Odd Even Linked List

Reversal Variants

These involve partial or grouped reversals, very common in interviews.

- 92. Reverse Linked List II
- 25. Reverse Nodes in k-Group (Hard but essential)
- 147. Insertion Sort List
- 148. Sort List

Merge & Divide

Key for recursive/merge thinking.

- 23. Merge k Sorted Lists
- 725. Split Linked List in Parts
- 2181. Merge Nodes in Between Zeros

Complex Logic / Design

Involves pointers + additional data structures or design decisions.

- 138. Copy List with Random Pointer
- 707. Design Linked List
- 146. LRU Cache
- 445. Add Two Numbers II
- 2. Add Two Numbers
- 382. Linked List Random Node
- 109. Convert Sorted List to Binary Search Tree

Recommended Strategy

- Practice pointer manipulation until it's second nature.
- Dry-run each problem on paper, especially cycle/intersection problems.
- Implement from scratch — no shortcuts via built-in structures unless allowed.
- Understand time/space tradeoffs for each method (iterative vs recursive, brute force vs optimized).

Focus Most On

- Cycle Detection (141, 142)
- Reversal Patterns (206, 92, 25)
- Merge & Sort (21, 23, 148)
- Random Pointer List (138)
- LRU Cache (146)
- Intersection Logic (160)
- Add Numbers (2, 445)
- Remove Nth Node from End (19)