Linked List

1. Explain the differences between arrays and linked lists in terms of:
    - Memory allocation
    - Access time
    - Insertion and deletion complexity
    - Use cases where one is preferred over the other

2. What are common memory-related bugs that can occur in linked list operations?

3. List at least 3 real-world problems or systems where linked lists are a better choice than arrays or other data structures. Justify your answer with reasoning focused on performance or flexibility.

4. You are given a singly linked list where each node contains an integer. Design an algorithm to determine whether the list contains a cycle.
    a) Describe your approach and analyze its time and space complexity.
    b) How would your approach differ if the list was doubly linked?

5. How would you detect a cycle in a singly linked list, and if one exists, remove the cycle without losing any nodes?

6. You're designing a memory-sensitive application. Would you choose a singly linked list, doubly linked list, or array? Justify your choice with respect to access patterns, memory usage, and insertion/deletion behavior.