

Problem Summary

Given the head of a singly linked list, reverse it so the last node becomes the head

Key Idea

We walk through the whole list and reverse each pointer

→ Instead of: $\text{curr} \rightarrow \text{next}$,

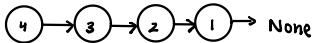
→ We do : $\text{curr} \rightarrow \text{prev}$

Diagram Example

Before:



After:



Approach

1) $\text{prev} = \text{None}$, $\text{curr} = \text{head}$

2) Save next node

3) Reverse current pointer

4) Move prev and curr forward

5) Repeat until end

6) Return prev

Time Complexity: $O(n)$ → each node is visited once

Space Complexity: $O(1)$ → in-place reversal