Detect cycle in Linked list II

```cpp
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    ListNode *detectCycle(ListNode *head) {
        ListNode* slow = head;
        ListNode* fast = head;

        while (fast && fast->next) {
            slow = slow->next;
            fast = fast->next->next;

            if (slow == fast) break;
        }

        if (!fast || !fast->next) return nullptr;

        fast = head;
        while (fast != slow) {
            fast = fast->next;
            slow = slow->next;
        }

        return slow;
    }
};
```