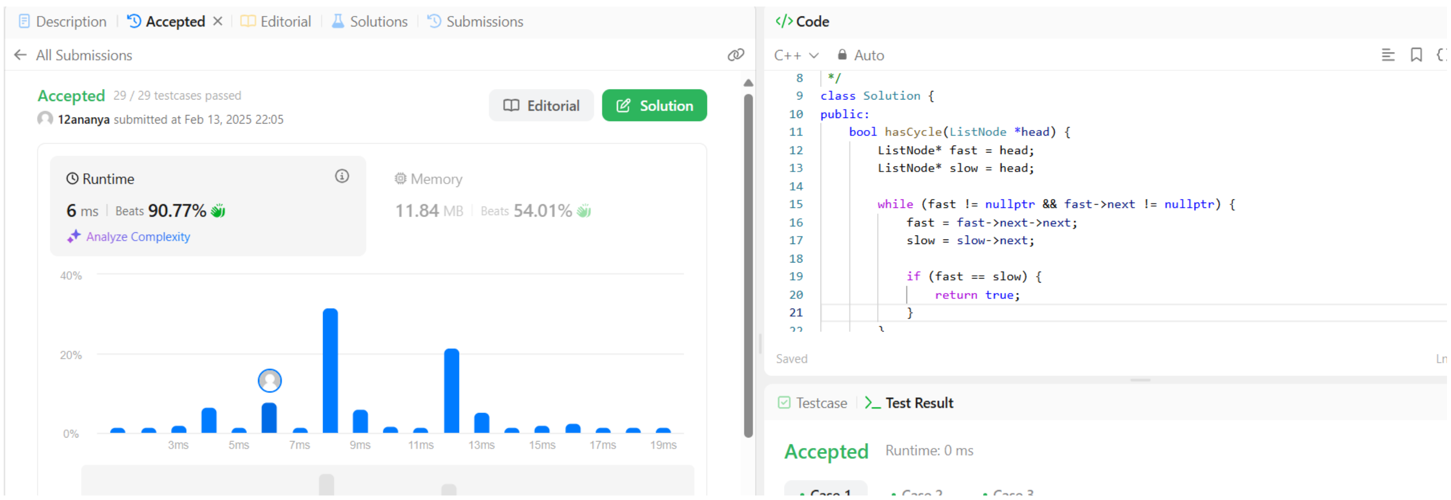
Linked list cycle

```cpp
class Solution {
public:
    bool hasCycle(ListNode *head) {
        ListNode* fast = head;
        ListNode* slow = head;

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

            if (fast == slow) {
                return true;
            }
        }

        return false;
    }
};
```



Accepted 29 / 29 testcases passed

12ananya submitted at Feb 13, 2025 22:05

Runtime
6 ms | Beats 90.77%

Memory
11.84 MB | Beats 54.01%

```cpp
 */
class Solution {
public:
    bool hasCycle(ListNode *head) {
        ListNode* fast = head;
        ListNode* slow = head;

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

            if (fast == slow) {
                return true;
            }
```

Accepted Runtime: 0 ms