

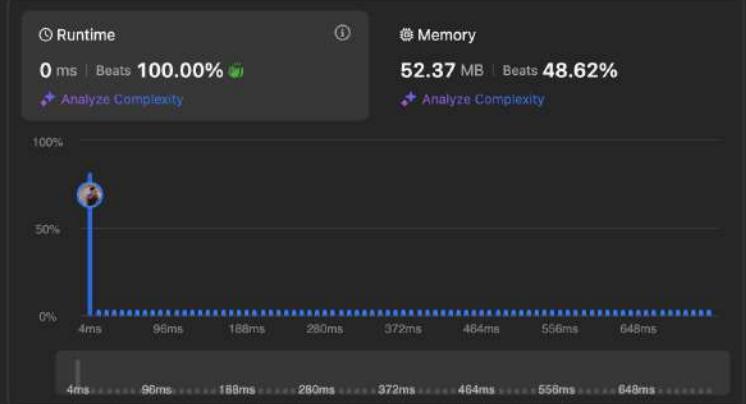
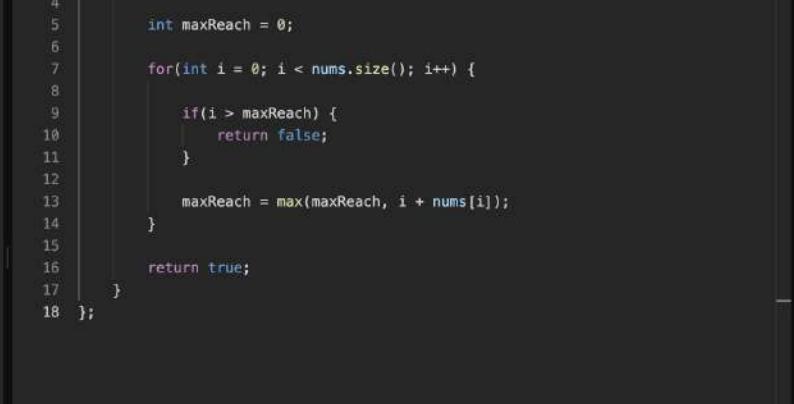
Problem List < >  Submit 

Description Accepted Editorial Solutions Submissions 

All Submissions 

Accepted 178 / 178 testcases passed

Siddhant0705 submitted at Feb 21, 2026 21:37

Runtime: 0 ms | Beats 100.00% 

Memory: 52.37 MB | Beats 48.62% 

Analyze Complexity

Code C++ 

```
1 class Solution {
2 public:
3     bool canJump(vector<int>& nums) {
4
5         int maxReach = 0;
6
7         for(int i = 0; i < nums.size(); i++) {
8
9             if(i > maxReach) {
10                 return false;
11             }
12
13             maxReach = max(maxReach, i + nums[i]);
14
15         }
16
17     }
18 }
```

Testcase | Test Result  Ln 18, Col 3

Accepted Runtime: 0 ms

Case 1 Case 2

Problem List < > Premium

Description Accepted Editorial Solutions Submissions

All Submissions

Accepted 51 / 51 testcases passed
Siddhant0705 submitted at Feb 21, 2026 21:39

Runtime 48 ms | Beats 83.69%
Memory 19.69 MB | Beats 90.39%
[Analyze Complexity](#)

Runtime distribution chart showing a single sharp peak at 10ms.

Code | C++

```
1 class Solution {
2 public:
3     int ladderLength(string beginWord, string endWord, vector<string>& wordList) {
4         unordered_set<string> st(wordList.begin(), wordList.end());
5         if(st.find(endWord) == st.end()) return 0;
6         queue<string> q;
7
8         for(int j = 0; j < word.length(); j++) {
9             char original = word[j];
10            for(char ch = 'a'; ch <= 'z'; ch++) {
11                word[j] = ch;
12                if(st.find(word) != st.end()) {
13                    q.push(word);
14                    st.erase(word);
15                }
16                word[j] = original;
17            }
18            level++;
19        }
20        return 0;
21    }
22 }
```

Saved Ln 45, Col 3

Testcase Test Result
Accepted Runtime: 0 ms
 Case 1 Case 2