

leetcode.com/problems/reverse-string/submissions/1832628535/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions | All Submissions

Accepted 477 / 477 testcases passed
7gZ8rRRYmV submitted at Nov 18, 2025 00:17

Runtime: 3 ms Beats 7.38% | Memory: 27.20 MB Beats 80.39%

Analyze Complexity

Code | C++

```
class Solution {
public:
    void reverseString(vector<char>& s) {
        int left = 0, right = s.size() - 1;
        while (left < right) {
            swap(s[left], s[right]);
            left++;
            right--;
        }
    }
};
```

Testcase | Test Result

leetcode.com/problems/design-twitter/submissions/1832609142/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 18 / 18 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:56

Runtime: 6 ms | Beats 69.94% | Analyze Complexity

Memory: 56.84 MB | Beats 24.62%

Code

```
C++ Auto
1 class Twitter {
2 private:
3     // userId -> set of followees
4     unordered_map<int, unordered_set<int>> follows;
5     // userId -> list of (timestamp, tweetId)
6     unordered_map<int, vector<pair<int,int>>> tweets;
7     // global timestamp (increasing)
8     int timeStamp;
9
10    struct Node {
11        int time;
12        int tweetId;
13        int user;
14        int idx; // index in that user's tweets vector
15    };
16
17    struct Cmp {
18        bool operator()(const Node& a, const Node& b) const {
19            return a.time < b.time; // max-heap by time
20        }
21    };
22
23 public:
24     Twitter() {
25         timeStamp = 0;
26     }
27
28     void postTweet(int userId, int tweetId) {

```

Testcase | Test Result

leetcode.com/problems/design-add-and-search-words-data-structure/submissions/1832587888/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 29 / 29 testcases passed
7gZBrRYmV submitted at Nov 17, 2025 23:34

Runtime: 315 ms | Beats 81.85% | Analyze Complexity

Memory: 563.13 MB | Beats 61.93%

Code | C++

```
1 class WordDictionary {
2     private:
3         struct Node {
4             bool isEnd;
5             Node* children[26];
6 
7             Node() {
8                 isEnd = false;
9                 for (int i = 0; i < 26; i++)
10                     children[i] = nullptr;
11             }
12         };
13 
14     Node* root;
15 
16 public:
17     WordDictionary() {
18         root = new Node();
19     }
20 
21     void addWord(string word) {
22         Node* curr = root;
23         for (char c : word) {
24             int idx = c - 'a';
25             if (!curr->children[idx])
26                 curr->children[idx] = new Node();
27             curr = curr->children[idx];
28         }
29     }
30 }
```

Testcase | Test Result

leetcode.com/problems/nth-digit/submissions/1832599286/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 72 / 72 testcases passed
7gZ8rRYmV submitted at Nov 17, 2025 23:46

Runtime: 0 ms | Beats 100.00% | Memory: 7.86 MB | Beats 64.62%

Analyze Complexity

Code | C++

```
class Solution {
public:
    int findNthDigit(int n) {
        long long length = 1;      // digit length (1-digit, 2-digit, ...)
        long long count = 9;       // number of integers with 'length' digits
        long long start = 1;       // starting integer for this digit-length

        // Step 1: Move across digit-length blocks
        while (n > length * count) {
            n -= length * count;
            length++;
            count *= 10;
            start *= 10;
        }

        // Step 2: Find the exact number containing the digit
        start += (n - 1) / length; // number containing the digit

        // Step 3: Find which digit inside the number
        string s = to_string(start);
        return s[(n - 1) % length] - '0';
    }
};
```

Saved

Testcase | Test Result

leetcode.com/problems/linked-list-random-node/submissions/1832605199/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions | Submit | Ctrl | Enter | Premium

All Submissions

Accepted 8 / 8 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:52

Runtime: 11 ms | Beats 6.37% | Memory: 23.19 MB | Beats 96.45%

Analyze Complexity

Runtime distribution chart:

Time Range (ms)	Percentage (%)
0-2	~22%
2-4	~5%
4-6	~18%
6-8	~5%
8-10	~5%
10-12	~5%

Code | C++

```
1 /**
2  * Definition for singly-linked list.
3  * struct ListNode {
4  *     int val;
5  *     ListNode *next;
6  * };
7 */
8
9 class Solution {
10 public:
11     ListNode* head;
12
13     Solution(ListNode* head) {
14         this->head = head;
15     }
16
17     int getRandom() {
18         int result = 0;
19         int count = 1;
20         ListNode* curr = head;
21
22         while (curr) {
23             // Reservoir Sampling step
24             if (rand() % count == 0) {
25                 result = curr->val;
26             }
27             curr = curr->next;
28             count++;
29         }
30     }
31 }
```

Testcase | Test Result

leetcode.com/problems/shuffle-an-array/submissions/1832606615/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 8 / 8 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:52

Runtime: 8 ms Beats 74.51% Memory: 63.68 MB Beats 43.63%

Analyze Complexity

Code C++ Auto

```
1 class Solution {
2 public:
3     vector<int> original;
4     vector<int> arr;
5
6     Solution(vector<int>& nums) {
7         original = nums;
8         arr = nums;
9         srand(time(NULL)); // seed RNG
10    }
11
12    vector<int> reset() {
13        arr = original;
14        return arr;
15    }
16
17    vector<int> shuffle() {
18        vector<int> res = arr;
19        int n = res.size();
20
21        for (int i = n - 1; i > 0; i--) {
22            int j = rand() % (i + 1);
23            swap(res[i], res[j]);
24        }
25
26        return res;
27    }
28}
```

Saved In 29, Col 1

Testcase | Test Result

leetcode.com/problems/custom-sort-string/submissions/1832585363/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 41 / 41 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:32

Runtime: 0 ms | Beats 100.00% | Memory: 8.25 MB | Beats 74.32%

Analyze Complexity

Code C++

```
class Solution {
public:
    string customSortString(string order, string s) {
        vector<int> freq(26, 0);

        // count frequency in s
        for (char c : s) {
            freq[c - 'a']++;
        }

        string result = "";

        // place characters according to order
        for (char c : order) {
            while (freq[c - 'a'] > 0) {
                result.push_back(c);
                freq[c - 'a']--;
            }
        }

        // place remaining characters
        for (char c = 'a'; c <= 'z'; c++) {
            while (freq[c - 'a'] > 0) {
                result.push_back(c);
                freq[c - 'a']--;
            }
        }
    }
}
```

Testcase | Test Result

leetcode.com/problems/implement-trie-prefix-tree/submissions/1832578630/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 16 / 16 testcases passed
7gZBrRYmV submitted at Nov 17, 2025 23:25

Runtime: 19 ms | Beats 71.00% | Memory: 50.52 MB | Beats 52.90%

Analyze Complexity

Runtime distribution chart showing performance across various execution times.

Code C++

```
1 class Trie {
2 private:
3     struct Node {
4         bool isEnd;
5         Node* children[26];
6     };
7     Node() {
8         isEnd = false;
9         for (int i = 0; i < 26; i++)
10             children[i] = nullptr;
11     }
12 };
13
14 Node* root;
15
16 public:
17     Trie() {
18         root = new Node();
19     }
20
21     void insert(string word) {
22         Node* curr = root;
23         for (char c : word) {
24             int idx = c - 'a';
25             if (!curr->children[idx])
26                 curr->children[idx] = new Node();
27             curr = curr->children[idx];
28         }
29     }
30 }
```

Saved

Ln 54, Col 1

Testcase | Test Result

Accepted | 49 / 49 testcases passed
7gZBrrRYmV submitted at Nov 17, 2025 23:49

Runtime: 80 ms | Beats 8.21% | Analyze Complexity

Memory: 38.81 MB | Beats 21.43%

```
class Solution {
public:
    bool isRectangleCover(vector<vector<int>>& rectangles) {
        long long areaSum = 0;
        int minX = INT_MAX, minY = INT_MAX;
        int maxX = INT_MIN, maxY = INT_MIN;

        unordered_set<string> corners;

        for (auto& r : rectangles) {
            int x1 = r[0], y1 = r[1];
            int x2 = r[2], y2 = r[3];

            // Update bounding box
            minX = min(minX, x1);
            minY = min(minY, y1);
            maxX = max(maxX, x2);
            maxY = max(maxY, y2);

            // Add area
            areaSum += 1LL * (x2 - x1) * (y2 - y1);

            // Count corners using a set
            string c1 = to_string(x1) + "," + to_string(y1);
            string c2 = to_string(x1) + "," + to_string(y2);
            string c3 = to_string(x2) + "," + to_string(y1);
            string c4 = to_string(x2) + "," + to_string(y2);
        }

        return areaSum == (maxX - minX) * (maxY - minY) && corners.size() == 4;
    }
}
```

leetcode.com/problems/group-anagrams/submissions/1832574110/?language=C%2B%2B

Problem List | Submit | Solutions | Submissions

Description | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 128 / 128 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:21

Runtime: 12 ms Beats 83.83% | Memory: 25.00 MB Beats 77.37%

Analyze Complexity

Runtime distribution chart showing execution times from 11ms to 78ms.

Code | C++

```
1 class Solution {
2 public:
3     vector<vector<string>> groupAnagrams(vector<string>& strs) {
4         unordered_map<string, vector<string>> mp;
5
6         for (string s : strs) {
7             string key = s;
8             sort(key.begin(), key.end()); // sorted string as key
9             mp[key].push_back(s);
10        }
11
12        vector<vector<string>> result;
13        for (auto &p : mp) {
14            result.push_back(p.second);
15        }
16
17        return result;
18    }
19};
```

Saved | Ln 20, Col 1

Testcase | Test Result

leetcode.com/problems/peeking-iterator/submissions/1832563126/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions | Submit | Ctrl | Enter | Premium

All Submissions

Accepted 15 / 15 testcases passed
7gZBrRYmV submitted at Nov 17, 2025 23:11

Runtime: 3 ms | Beats 54.16% | Analyze Complexity

Memory: 12.24 MB | Beats 19.33%

Runtime distribution: 2ms, 3ms, 4ms, 5ms

Code: C++

```
1 /**
2 * Definition for the Iterator interface.
3 * class Iterator {
4 *     struct Data;
5 *     Data* data;
6 *
7 * public:
8 *     Iterator(const vector<int>& nums);
9 *     Iterator(const Iterator& iter);
10 *
11 *     int next();
12 *     bool hasNext() const;
13 * };
14
15 class PeekingIterator : public Iterator {
16 private:
17     bool hasPeeked;
18     int peekVal;
19
20 public:
21     PeekingIterator(const vector<int>& nums) : Iterator(nums) {
22         hasPeeked = false;
23         peekVal = 0;
24     }
25
26     int peek() {
27         if (!hasPeeked) {
28             peekVal = Iterator::next();
```

Testcase | Test Result

leetcode.com/problems/isomorphic-strings/submissions/183257206/language-C%2B%2B

Problem List | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 47 / 47 test cases passed

7020RRYmV submitted at Nov 17, 2025 23:20

Runtime: 0 ms | Beats 100.00% | Memory: 9.42 MB | Effic 27.93%

Runtime Analysis Component

Code C++

```
1 class Solution {
2 public:
3     bool isIsomorphic(string s, string t) {
4         if (s.size() != t.size()) return false;
5
6         unordered_map<char, char> mp1;
7         unordered_map<char, char> mp2;
8
9         for (int i = 0; i < s.size(); i++) {
10             char a = s[i];
11             char b = t[i];
12
13             if (mp1.count(a) && mp1[a] != b) return false;
14             if (mp2.count(b) && mp2[b] != a) return false;
15
16             mp1[a] = b;
17             mp2[b] = a;
18         }
19
20     }
21 };
22 }
```

Saved

Test Result

leetcode.com/problems/ransom-note/submissions/1832567443/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 130 / 130 testcases passed
7g28RRYmV submitted at Nov 17, 2025 23:15

Runtime: 0 ms | Beats 100.00% | Memory: 11.66 MB | Beats 89.45%

Analyze Complexity

Runtime distribution chart showing performance across various execution times.

Code | C++

```
class Solution {
public:
    bool canConstruct(string ransomNote, string magazine) {
        vector<int> freq(26, 0);

        for (char c : magazine)
            freq[c - 'a']++;

        for (char c : ransomNote) {
            if (freq[c - 'a'] == 0)
                return false;
            freq[c - 'a']--;
        }

        return true;
    }
};
```

Saved | Ln 18, Col 1

Testcase | Test Result

leetcode.com/problems/complex-number-multiplication/submissions/1832548829/?language=C%2B%2B

Problem List < > ⌂

Description Accepted Note Editorial Solutions Submissions Submit Ctrl Enter ⌂

All Submissions

Accepted 56 / 56 testcases passed
7gZ8eRRYmV submitted at Nov 17, 2025 22:58

Runtime 0 ms | Beats 100.00% Memory 8.00 MB | Beats 51.70%

Analyze Complexity

100% 50% 0%

Code | C++

```
class Solution {
public:
    string complexNumberMultiply(string num1, string num2) {
        auto p1 = parse(num1);
        auto p2 = parse(num2);

        int a = p1.first, b = p1.second;
        int c = p2.first, d = p2.second;

        int real = a * c - b * d;
        int imag = a * d + b * c;

        return to_string(real) + "+" + to_string(imag) + "i";
    }

    pair<int, int> parse(string s) {
        int plusPos = s.find('+');
        int iPos = s.find('i');

        int real = stoi(s.substr(0, plusPos));
        int imag = stoi(s.substr(plusPos + 1, iPos - plusPos - 1));

        return {real, imag};
    }
};
```

Saved Ln 26, Col 1

Testcase Test Result

leetcode.com/problems/multiply-strings/submissions/1832552461/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 311 / 311 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:01

Runtime: 4 ms | Beats 31.13%
Memory: 9.62 MB | Beats 54.66%

Runtime Analysis: A histogram showing execution time distribution. The x-axis ranges from 5ms to 25ms, and the y-axis ranges from 0% to 30%. The distribution is highly skewed, with most executions taking between 5ms and 10ms.

Code C++

```
class Solution {
public:
    string multiply(string num1, string num2) {
        if (num1 == "0" || num2 == "0") return "0";
        int n1 = num1.size(), n2 = num2.size();
        vector<int> result(n1 + n2, 0);

        for (int i = n1 - 1; i >= 0; i--) {
            for (int j = n2 - 1; j >= 0; j--) {
                int mul = (num1[i] - '0') * (num2[j] - '0');

                result[i + j + 1] = sum % 10;
                result[i + j] += sum / 10;
            }
        }

        string ans = "";
        int i = 0;

        // Skip leading zeros (if any)
        while (i < result.size() && result[i] == 0) i++;

        for (; i < result.size(); i++)
            ans.push_back(result[i] + '0');

        return ans;
    }
}
```

Testcase | Test Result

leetcode.com/problems/valid-palindrome/submissions/1832557248?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 487 / 487 testcases passed
7gZ8rRYYmV submitted at Nov 17, 2025 23:05

Runtime: 6 ms Beats 8.00% | Memory: 10.06 MB Beats 42.03%

Analyze Complexity

Code

C++ v Auto

```
1 class Solution {
2 public:
3     bool isPalindrome(string s) {
4         int left = 0, right = s.size() - 1;
5
6         while (left < right) {
7             while (left < right && !isalnum(s[left])) left++;
8             while (left < right && !isalnum(s[right])) right--;
9
10            if (tolower(s[left]) != tolower(s[right]))
11                return false;
12
13            left++;
14            right--;
15        }
16
17        return true;
18    }
19};
```

Testcase | Test Result

leetcode.com/problems/min-stack/submissions/1832547479/?language=C%2B%2B

Problem List | Description | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 32 / 32 testcases passed
7gZ8rRYmV submitted at Nov 17, 2025 22:56

Runtime: 1 ms Beats 73.67% | Memory: 23.23 MB Beats 79.08%

Analyze Complexity



The image shows two bar charts. The left chart, titled 'Runtime', compares execution times from 1ms to 10ms. The right chart, titled 'Memory', compares memory usage from 0MB to 30MB. Both charts show the percentage of submissions that beat the current submission.

Code | C++

```
class MinStack {
public:
    stack<long long> st;
    long long minVal;
```

Code

```
1 class MinStack {
2     public:
3         stack<long long> st;
4         long long minVal;
5
6     MinStack() {
7         minVal = LLONG_MAX;
8     }
9
10    void push(int val) {
11        if (st.empty()) {
12            st.push(val);
13            minVal = val;
14        } else {
15            if (val < minVal) {
16                // Encode value
17                st.push(2LL * val - minVal);
18                minVal = val;
19            } else {
20                st.push(val);
21            }
22        }
23    }
24
25    void pop() {
26        long long topVal = st.top();
27        st.pop();
28        if (topVal < minVal) {
```

Saved

Ln 46, Col 1

Testcase | Test Result

leetcode.com/problems/design-underground-system/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 57 / 57 testcases passed
7g28rRYYmV submitted at Nov 17, 2025 22:54

Runtime: 23 ms | Beats 82.39% | Memory: 68.08 MB | Beats 59.74%

Analyze Complexity

Runtime Performance Chart (ms):

Code | C++

```
class UndergroundSystem {  
public:  
    // Stores: id -> {stationName, time}  
    unordered_map<int, pair<string, int>> checkInMap;  
    // Stores: "startStation endStation" -> {totalTime, count}  
    unordered_map<string, pair<long long, int>> travelData;  
    UndergroundSystem() {}  
    void checkIn(int id, string stationName, int t) {  
        checkInMap[id] = {stationName, t};  
    }  
    void checkOut(int id, string stationName, int t) {  
        auto &entry = checkInMap[id];  
        string startStation = entry.first;  
        int startTime = entry.second;  
        string routeKey = startStation + "_" + stationName;  
        int travelTime = t - startTime;  
        travelData[routeKey].first += travelTime;  
        travelData[routeKey].second++;  
        checkInMap.erase(id);  
    }  
};
```

Saved | Ln 35, Col 1

Testcase | Test Result

leetcode.com/problems/add-strings/submissions/1832550245?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 317 / 317 testcases passed
7gZBrRYmV submitted at Nov 17, 2025 22:59

Runtime: 0 ms | Beats 100.00% | Analyze Complexity

Memory: 9.10 MB | Beats 79.05% | Analyze Complexity

Runtime distribution chart showing 99.9% of submissions between 0ms and 5ms.

Code: C++

```
1 class Solution {
2 public:
3     string addStrings(string num1, string num2) {
4         int i = num1.size() - 1;
5         int j = num2.size() - 1;
6         int carry = 0;
7         string result = "";
8
9         while (i >= 0 || j >= 0 || carry) {
10             int sum = carry;
11
12             if (i >= 0) sum += num1[i--] - '0';
13             if (j >= 0) sum += num2[j--] - '0';
14
15             result.push_back((sum % 10) + '0');
16             carry = sum / 10;
17         }
18
19         reverse(result.begin(), result.end());
20         return result;
21     }
22 };
```

Saved

Testcase | Test Result

leetcode.com/problems/string-compression/submissions/1832556170/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions | All Submissions

Accepted 77 / 77 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:04

Runtime: 0 ms | Beats 100.00% | Analyze Complexity

Memory: 13.86 MB | Beats 44.87%

2.85% of solutions used 2 ms of runtime

Code | C++

```
class Solution {
public:
    int compress(vector<char>& chars) {
        int n = chars.size();
        int write = 0; // position to write compressed chars
```

Code | C++

```
1 class Solution {
2 public:
3     int compress(vector<char>& chars) {
4         int n = chars.size();
5         int write = 0; // position to write compressed chars
6         int read = 0; // position to read chars
7
8         while (read < n) {
9             char current = chars[read];
10            int count = 0;
11
12            // count occurrences
13            while (read < n && chars[read] == current) {
14                read++;
15                count++;
16            }
17
18            // write character
19            chars[write++] = current;
20
21            // write count if > 1
22            if (count > 1) {
23                string cnt = to_string(count);
24                for (char c : cnt) {
25                    chars[write++] = c;
26                }
27            }
28        }
29    }
30}
```

Testcase | Test Result

Problem List < > ✎

Description | Accepted X | Note X | Editorial | Solutions | Submissions

All Submissions

Accepted 102 / 102 testcases passed
7gZ8rRYmV submitted at Nov 17, 2025 19:47

Editorial Solution

Runtime: 8 ms | Beats 45.93% | Analyze Complexity

Memory: 38.80 MB | Beats 50.00%

2.88% of solutions used 15 ms of runtime

Code | C++

```
class ParkingSystem {
public:
    int slots[4]; // index 1: big, 2: medium, 3: small
```

C++ v Auto

```
1 class ParkingSystem {
2 public:
3     int slots[4]; // index 1: big, 2: medium, 3: small
4
5     ParkingSystem(int big, int medium, int small) {
6         slots[1] = big;
7         slots[2] = medium;
8         slots[3] = small;
9     }
10
11    bool addCar(int carType) {
12        if (carType < 1 || carType > 3) return false;
13        if (slots[carType] > 0) {
14            --slots[carType];
15            return true;
16        }
17        return false;
18    }
19
20 }
```

Saved

leetcode.com/problems/subrectangle-queries/submissions/1832531640/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 52 / 52 testcases passed
7gZBrRyMv submitted at Nov 17, 2025 22:42

Runtime: 28 ms | Beats 54.78% | Analyze Complexity

Memory: 25.38 MB | Beats 78.09% | Analyze Complexity

Code C++

```
1 class SubrectangleQueries {
2 public:
3     vector<vector<int>> rect;
4
5     SubrectangleQueries(vector<vector<int>>& rectangle) {
6         rect = rectangle;
7     }
8
9     void updateSubrectangle(int row1, int col1, int row2, int col2, int newValue) {
10        for (int i = row1; i <= row2; i++) {
11            for (int j = col1; j <= col2; j++) {
12                rect[i][j] = newValue;
13            }
14        }
15    }
16
17    int getValue(int row, int col) {
18        return rect[row][col];
19    }
20};
```

Saved | Ln 21, Col 1

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

leetcode.com/problems/word-pattern/submissions/1832572147/?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 44 / 44 testcases passed
7gZBRRYmV submitted at Nov 17, 2025 23:19

Runtime: 0 ms | Beats 100.00% | Memory: 8.78 MB | Beats 69.82%

Analyze Complexity

Runtime Analysis: A large blue bar indicates the main execution time, while smaller bars below show overhead components like memory allocation and deallocation.

Code | C++

```
class Solution {
public:
    bool wordPattern(string pattern, string s) {
        vector<string> words;
        string temp;
        // split s into words
        while (ss >> temp) {
            words.push_back(temp);
        }
        if (words.size() != pattern.size())
            return false;
        unordered_map<char, string> mp1;
        unordered_map<string, char> mp2;
        for (int i = 0; i < pattern.size(); i++) {
            char c = pattern[i];
            string w = words[i];
            if (mp1.count(c) && mp1[c] != w)
                return false;
            if (mp2.count(w) && mp2[w] != c)
                return false;
            mp1[c] = w;
            mp2[w] = c;
        }
    }
}
```

Testcase | Test Result

leetcode.com/problems/number-of-recent-calls/submissions/1832561768?language=C%2B%2B

Problem List | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 68 / 68 testcases passed
7gZ8rRRYmV submitted at Nov 17, 2025 23:09

Runtime: 18 ms | Beats 53.38% | Analyze Complexity

Memory: 64.24 MB | Beats 37.58%

Runtime Distribution:

Memory Distribution:

Code

C++ Auto

```
1 class RecentCounter {
2 public:
3     queue<int> q;
4
5     RecentCounter() {}
6
7     int ping(int t) {
8         q.push(t);
9
10        // Remove pings older than 3000 ms
11        while (!q.empty() && q.front() < t - 3000) {
12            q.pop();
13        }
14
15        return q.size();
16    }
17}
18
```

Saved

Ln 18, Col 1

Testcase | Test Result

leetcode.com/problems/time-based-key-value-store/submissions/1832559199/?language=C%2B%2B

Click to go back, hold to see history

Problem List | Submit | Enter | Premium

Description | Accepted | Note | Editorial | Solutions | Submissions

All Submissions

Accepted 54 / 54 testcases passed

7gZ8rRYmV submitted at Nov 17, 2025 23:07

Runtime: 67 ms | Beats 27.33% | Analyze Complexity

Memory: 136.82 MB | Beats 62.18%

Code C++

```
class TimeMap {
public:
    unordered_map<string, vector<pair<int, string>> mp;
    TimeMap() {}
    void set(string key, string value, int timestamp) {
        mp[key].push_back({timestamp, value});
    }
    string get(string key, int timestamp) {
        if (!mp.count(key)) return "";
        auto &vec = mp[key];
        int left = 0, right = vec.size() - 1;
        string ans = "";
        // binary search for <= timestamp
        while (left <= right) {
            int mid = left + (right - left) / 2;
            if (vec[mid].first <= timestamp) {
                ans = vec[mid].second;
                left = mid + 1;
            } else {
                right = mid - 1;
            }
        }
        return ans;
    }
};
```

Testcase | Test Result