

← All Submissions

Accepted 88 / 88 testcases passed

 **SHREYA RAJ** submitted at Feb 18, 2026 15:30

 Editorial

 Solution

 Runtime

0 ms Beats **100.00%** 

 Analyze Complexity



Memory

48.55 MB Beats 11.88%






Java  Auto

```
28 |         left = mid + 1;
29 |     }
30 | }
31 |
32 |
33 |     return idx;
34 | }
35 |
36 | }
```

Saved

Ln 36, Col 2


☒ Testcase  Test Result

Accepted Runtime: 0 ms

☒ Case 1 ☒ Case 2 ☒ Case 3

← All Submissions

Accepted 196 / 196 testcases passed

 **SHREYA RAJ** submitted at Feb 18, 2026 15:28

 Editorial

 Solution

⌚ Runtime

0 ms Beats **100.00%** 

 Analyze Complexity

ⓘ

💾 Memory



43.92 MB Beats **45.56%**





Java ▾

 Auto

☰   ↶ ↷ ↵

```
20         } else {  
21             right = mid - 1;  
22         }  
23     }  
24 }  
25  
26     return -1;  
27 }  
28 }  
29
```

Saved

Ln 28, Col 2


☒ Testcase  **Test Result**

Accepted Runtime: 0 ms

☒ Case 1 ☒ Case 2 ☒ Case 3

← All Submissions

Accepted 196 / 196 testcases passed

 **SHREYA RAJ** submitted at Feb 18, 2026 15:26

 Editorial

 Solution

⌚ Runtime

0 ms Beats **100.00%** 

 Analyze Complexity

ⓘ

💾 Memory

43.88 MB Beats **65.68%** 





Java ▾

 Auto

☰ 📖 ⌕ ↺ ↻ ↶ ↷

```
20         } else {  
21             right = mid - 1;  
22         }  
23     }  
24 }  
25  
26     return -1;  
27 }  
28 }
```

Saved

Ln 28, Col 2

☒ Testcase  Test Result

Accepted Runtime: 0 ms

☒ Case 1 ☒ Case 2 ☒ Case 3

Accepted 294 / 294 testcases passed

SHREYA RAJ submitted at Feb 18, 2026 15:25

Editorial

Solution

Runtime

19 ms Beats 80.46%

Analyze Complexity

Memory

45.96 MB

Beats 46.98%



Code Java

```
1 class Solution {
2     public List<List<Integer>> fourSum(int[] nums, int target) {
3         List<List<Integer>> answer = new ArrayList<>();
4         Arrays.sort(nums);
```

Code

Java Auto

```
33         left++;
34         right--;
35     }
36     }
37     }
38     }
39     return answer;
40 }
41 }
```

Saved

Ln 41, Col 2

Testcase Test Result

Accepted Runtime: 1 ms

Case 1 Case 2

Input

nums =
[1,0,-1,0,-2,2]

target =
0

Output

All Submissions

Accepted 88 / 88 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:24

Editorial Solution

Runtime

286 ms Beats 9.01%

Analyze Complexity



Memory

47.48 MB Beats 6.85%



Code Java

```
1 class Solution {
2     private int rows;
3     private int cols;
4     private Set<String> visited;
5 }
```

Code

Java Auto

```
40     visited.add(r + "," + c);
41     boolean res = dfs(board, word, r + 1, c, k + 1) ||
42                 dfs(board, word, r - 1, c, k + 1) ||
43                 dfs(board, word, r, c + 1, k + 1) ||
44                 dfs(board, word, r, c - 1, k + 1);
45     visited.remove(r + "," + c);
46     return res;
47 }
48 }
```

Saved

Ln 48, Col 2

Testcase Test Result

Accepted Runtime: 3 ms

Case 1 Case 2 Case 3

Input

board =
[["A","B","C","E"],["S","F","C","S"],["A","D","E","E"]]

word =
"ABCCED"

Output

True

Description Accepted x Editorial Solutions Submissions

All Submissions

Accepted 10 / 10 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:22

Editorial Solution

Runtime

1 ms Beats 87.92%
Analyze Complexity

i

Memory

44.26 MB Beats 57.90%



Code Java

```
1 class Solution {  
2     public List<List<Integer>> subsets(int[] nums) {  
3         List<List<Integer>> res = new ArrayList<>();  
4         List<Integer> subset = new ArrayList<>();  
5     }
```

Code

Java Auto

```
14     }  
15  
16     subset.add(nums[index]);  
17     createSubset(nums, index + 1, res, subset);  
18  
19     subset.remove(subset.size() - 1);  
20     createSubset(nums, index + 1, res, subset);  
21 }  
22 }
```

Saved

Ln 22, Col 2

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

nums =
[1,2,3]

Output

[[1,2,3],[1,2],[1,3],[1],[2,3],[2],[3],[]]

Expected

Description Accepted x Editorial Solutions Submissions

All Submissions

Accepted 89 / 89 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:14

Editorial Solution

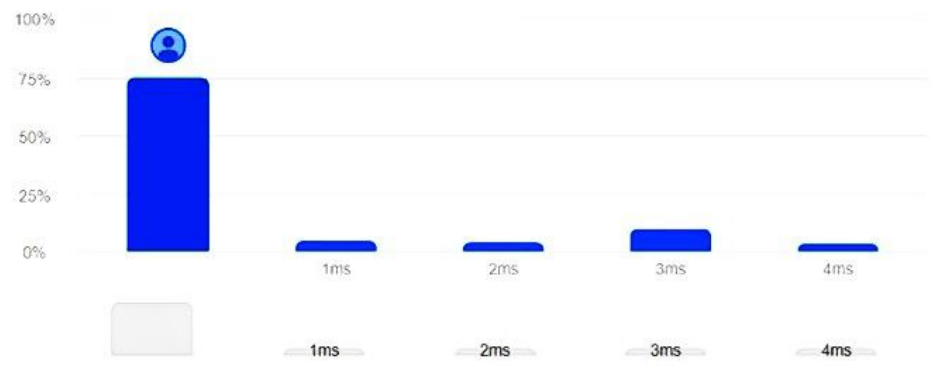
Runtime

0 ms Beats 100.00%

Analyze Complexity

Memory

43.61 MB Beats 42.12%



Code Java

```
1 class Solution {  
2     public void sortColors(int[] nums) {  
3  
4         int r = 0, w = 0, b = nums.length-1;  
5
```

Code

Java Auto

```
18         nums[b] = temp;  
19         b--;  
20  
21     }  
22  
23     }  
24  
25 }  
26 }
```

Saved

Ln 26, Col 2

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

nums =
[2,0,2,1,1,0]

Output

[0,0,1,1,2,2]

Expected

All Submissions

Accepted 133 / 133 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:13

Editorial Solution

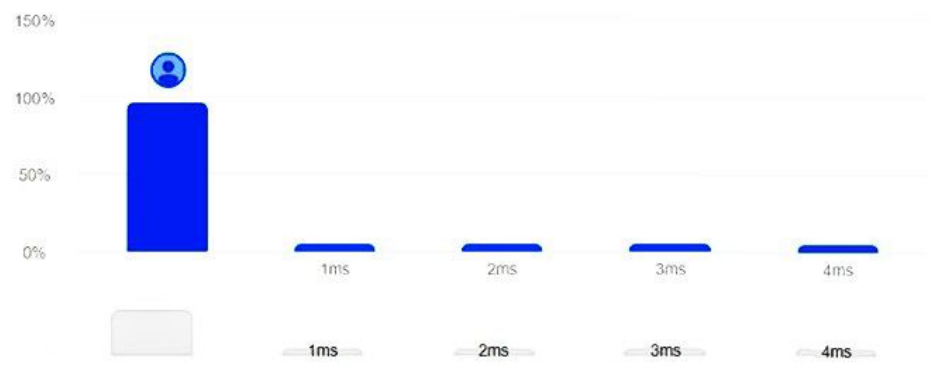
Runtime

0 ms Beats 100.00%
Analyze Complexity



Memory

44.06 MB Beats 42.20%



Code Java

```
1
2 class Solution {
3     public boolean searchMatrix(int[] [] matrix, int target)
4     {
5         int iRow = matrix.length;
```

Code

Java Auto

```
28
29
30
31
32
33
34
35
36
    {
        low = mid + 1;
    }
}
return false;
```

Saved

Ln 36, Col 2

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

matrix =
[[1, 3, 5, 7], [10, 11, 16, 20], [23, 30, 34, 60]]

target =
3

Output

True

All Submissions

Accepted 114 / 114 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:09

Editorial Solution

Runtime

0 ms Beats 100.00%

Analyze Complexity



Memory

43.41 MB Beats 57.87%



Code Java

```
1 class Solution {
2     public int[] plusOne(int[] digits) {
3         for (int i = digits.length - 1; i >= 0; i--) {
4             if (digits[i] + 1 != 10) {
5                 digits[i] += 1;
```

Code

Java Auto

```
7         }
8         digits[i] = 0;
9     }
10
11     int[] newDigits = new int[digits.length + 1];
12     newDigits[0] = 1;
13     return newDigits;
14 }
15 }
```

Saved

Ln 15, Col 2

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

digits =
[1,2,3]

Output

[1,2,4]

Expected

All Submissions

Accepted 128 / 128 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:08

Editorial Solution

Runtime

6 ms Beats 99.36%



Memory

49.72 MB Beats 62.96%

Code

```
1 class Solution {
2     public List<List<String>> groupAnagrams(String[] strs) {
3         Map<String, List<String>> map = new HashMap<>();
4
5         for (String word : strs) {
```

Code

```
Java Auto
12     }
13
14     map.get(sortedWord).add(word);
15 }
16
17 return new ArrayList<>(map.values());
18 }
19 }
20 }
```

Saved

Ln 20, Col 1

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

strs =
["eat","tea","tan","ate","nat","bat"]

Output

[["eat","tea","ate"],["bat"],["tan","nat"]]

Expected

← All Submissions

Accepted 110 / 110 testcases passed
 SHREYA RAJ submitted at Feb 18, 2026 15:05

[Editorial](#)
[Solution](#)

Runtime
 Memory
 1 ms Beats 99.56%
 47.46 MB Beats 36.29%
 Analyze Complexity



Code
 Java

```

1 class Solution {
2     public int jump(int[] nums) {
3         int near = 0, far = 0, jumps = 0;
4
5         while (far < nums.length - 1) {

```

Code

```

Java
Auto
9     }
10    near = far + 1;
11    far = farthest;
12    jumps++;
13    }
14
15    return jumps;
16 }
17 }
```

Saved
 Ln 17, Col 2

Testcase
 Test Result

Accepted Runtime: 0 ms

Case 1
 Case 2

Input
 nums =
 [2,3,1,1,4]

Output
 2

Expected

Description Editorial Solutions Accepted x Submissions

All Submissions

Accepted 176 / 176 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:04

Editorial Solution

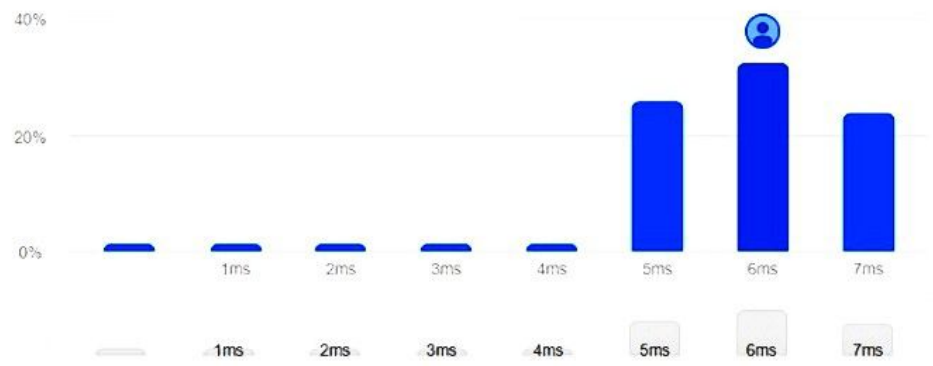
Runtime

6 ms Beats 73.17%

Analyze Complexity

Memory

45.06 MB Beats 87.81%



Code Java

```
1 class Solution {  
2     public List<List<Integer>> combinationSum2(int[] candidates, int target)  
3     {  
4         Arrays.sort(candidates);  
5         List<List<Integer>> res = new ArrayList<>();  
6     }  
7 }
```

Code

Java Auto

```
26         break;  
27     }  
28  
29     comb.add(candidates[i]);  
30     dfs(candidates, target - candidates[i], i + 1, comb, res);  
31     comb.remove(comb.size() - 1);  
32 }  
33  
34 }
```

Saved

Ln 34, Col 2

Testcase Test Result

Accepted Runtime: 1 ms

Case 1 Case 2

Input

candidates =
[10,1,2,7,6,1,5]

target =
8

Output

[[1, 1, 6], [1, 2, 5], [1, 7], [2, 6]]

Description Editorial Solutions Accepted x Submissions

All Submissions

Accepted 160 / 160 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 15:01

Editorial Solution

Runtime

3 ms Beats 36.93%

Analyze Complexity

Memory

45.96 MB Beats 34.02%



Code

```
1 class Solution {  
2     public List<List<Integer>> combinationSum(int[] candidates, int target)  
3     {  
4         List<List<Integer>> res = new ArrayList<>();  
5         makeCombination(candidates, target, 0, new ArrayList<>(), res);  
6     }  
7 }
```

Code

Java Auto

```
16     return;  
17 }  
18  
19     comb.add(candidates[idx]);  
20     makeCombination(candidates, target, idx, comb, total + candidates[idx], res);  
21     comb.remove(comb.size() - 1);  
22     makeCombination(candidates, target, idx + 1, comb, total, res);  
23 }  
24 }
```

Saved

Ln 24, Col 2

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

candidates =
[2,3,6,7]

target =
7

Output

All Submissions

Accepted 66 / 66 testcases passed
SHREYA RAJ submitted at Feb 18, 2026 14:59

Editorial Solution

Runtime

0 ms Beats 100.00%

Analyze Complexity

Memory

44.72 MB Beats 66.75%



Code Java

```
1 class Solution {
2     public int searchInsert(int[] nums, int target) {
3         int left = 0;
4         int right = nums.length - 1;
5     }
```

Submit Ctrl Enter

Java Auto

```
12         right = mid - 1;
13     } else {
14         left = mid + 1;
15     }
16 }
17
18 return left;
19 }
20 }
```

Saved

Ln 20, Col 2

Testcase Test Result

Accepted Runtime: 0 ms

Case 1 Case 2 Case 3

Input

nums =
[1,3,5,6]

target =
5

Output