

# 第五周练习

张睿恒 2024302182001

完成题目：LeetCode167（简单），LeetCode202（简单），LeetCode49（中等），  
LeetCode18（中等），LeetCode36（中等），LeetCode90（困难）

## T1 LeetCode167 简单

```
class Solution {  
public:  
    vector<int> twoSum(vector<int>& numbers, int target) {  
        int l = 0;  
        int r = numbers.size()-1;  
        int now = numbers[l] + numbers[r];  
        while(now != target){  
            if(now >= target) --r;  
            else ++l;  
            now = numbers[l] + numbers[r];  
        }  
        return {l+1,r+1};  
    }  
};
```

Lee × | Accepted ×

← All Submissions



Accepted 24 / 24 testcases passed

Itheng submitted at Nov 05, 2025 18:11

Editorial

Solution

Runtime



0 ms | Beats 100.00%

Analyze Complexity

Memory

19.53 MB | Beats 31.78%



Code | C++

```
class Solution {
public:
    vector<int> twoSum(vector<int>& numbers, int target) {
        int l = 0;
        int r = numbers.size() - 1;
        int now = numbers[l] + numbers[r];
        while(now != target){
            if(now >= target) --r;
            else ++l;
            now = numbers[l] + numbers[r];
        }
    }
}
```

View more

## T2 LeetCode202 简单

```
class Solution {  
public:  
    bool isHappy(int n) {  
        int s = check(n);  
        int f = check(check(n));  
        while(s != f){  
            s = check(s);  
            f = check(check(f));  
        }  
  
        return s==1?true:false;  
    }  
  
private:  
    int check(int x){  
        int sum = 0;  
        while(x>0){  
            sum += pow(x%10,2);  
            x /= 10;  
        }  
        return sum;  
    }  
};
```

Lee × | Accepted ×

← All Submissions



Accepted 420 / 420 testcases passed

Itheng submitted at Nov 05, 2025 18:21

Editorial

Solution

⌚ Runtime



0 ms | Beats 100.00%

Analyze Complexity

🐏 Memory

7.66 MB | Beats 96.97%



Code | C++

```
class Solution {
public:
    bool isHappy(int n) {
        int s = check(n);
        int f = check(check(n));
        while(s != f){
            s = check(s);
            f = check(check(f));
        }
    }
}
```

View more

# T3 Leetcode49 中等

```
class Solution {
public:
    vector<vector<string>> groupAnagrams(vector<string>& strs) {
        vector < vector <string> > ans;
        unordered_map < string , vector <string> > map1;

        for(int i=0;i<strs.size();++i){
            string key = strs[i];
            sort(key.begin(),key.end());
            map1[key].push_back(strs[i]);
        }

        for(unordered_map<string, vector<string>>::iterator it =
map1.begin();it!=map1.end();++it) ans.push_back(it->second);

        return ans;
    }
};
```

Lee × | Accepted ×

← All Submissions



Accepted 128 / 128 testcases passed

Itheng submitted at Nov 05, 2025 18:30

Editorial

Solution

Runtime

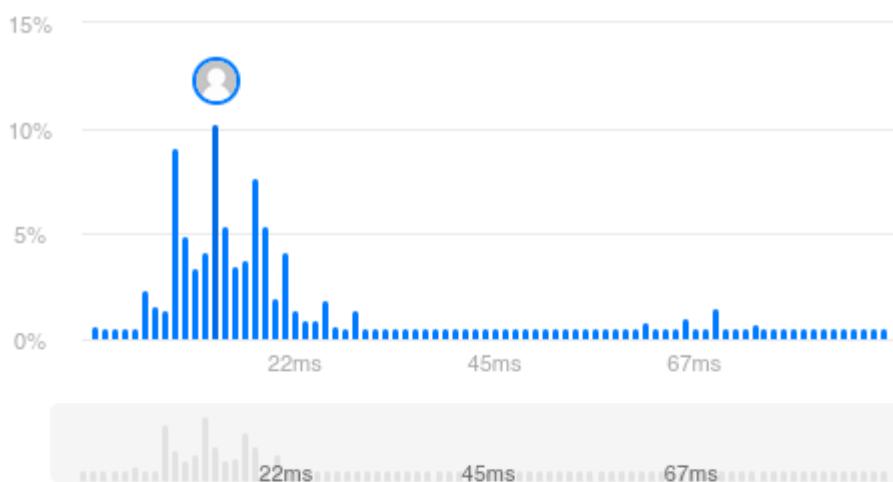


14 ms | Beats 75.51%

Analyze Complexity

Memory

25.00 MB | Beats 77.25%



Code | C++

```
class Solution {
public:
    vector<vector<string>> groupAnagrams(vector<string>& s1)
    {
        vector < vector <string> > ans;
        unordered_map < string , vector <string> > map1;

        for(int i=0;i<s1.size();++i){
            string key = s1[i];
            ↴ View more
        }
    }
};
```

T4 Leetcode18 中等

```

typedef long long ll;

class Solution {
public:
    vector<vector<int>> fourSum(vector<int>& nums, int target) {
        int n = nums.size();
        set<vector <int> > uniqueRes;

        for (int i = 0; i < n; i++) {
            for (int j = i + 1; j < n; j++) {
                for (int k = j + 1; k < n; k++) {
                    ll sum3 = (ll)nums[i] + nums[j] + nums[k];
                    ll missing = 1ll*target - sum3;

                    for (int m = 0; m < n; m++) {
                        if (m != i && m != j && m != k && nums[m] == missing) {
                            vector <int> temp = {nums[i], nums[j], nums[k],
(int)missing};
                            sort(temp.begin(), temp.end());
                            uniqueRes.insert(temp);
                            break;
                        }
                    }
                }
            }
        }

        return vector < vector <int> >(uniqueRes.begin(), uniqueRes.end());
    }
};

```

Lee × | Accepted ×

← All Submissions



Accepted 294 / 294 testcases passed

Itheng submitted at Nov 05, 2025 18:42

Editorial

Solution

Runtime



1394 ms | Beats 5.00%

Analyze Complexity

Memory

104.12 MB | Beats 5.17%



Code | C++

```
typedef long long ll;

class Solution {
public:
    vector<vector<int>> fourSum(vector<int>& nums, int target) {
        int n = nums.size();
        set<vector <int> > uniqueRes;
        sort(nums.begin(), nums.end());
        for (int i = 0; i < n - 3; i++) {
            for (int j = i + 1; j < n - 2; j++) {
                int left = j + 1;
                int right = n - 1;
                while (left < right) {
                    if (nums[i] + nums[j] + nums[left] + nums[right] == target) {
                        uniqueRes.insert({nums[i], nums[j], nums[left], nums[right]});
                        left++;
                        right--;
                    } else if (nums[i] + nums[j] + nums[left] + nums[right] < target) {
                        left++;
                    } else {
                        right--;
                    }
                }
            }
        }
        return uniqueRes;
    }
};
```

View more

# T5 LeetCode36 中等

```
class Solution {
public:
    bool isValidSudoku(vector < vector <char> & board) {
        unordered_set <string> seen;

        for (int i = 0; i < 9; ++i)
            for (int j = 0; j < 9; ++j) {
                if (board[i][j] == '.') continue;
                const string c(1, board[i][j]);
                if (!seen.insert(c + "@row" + to_string(i)).second or
                    !seen.insert(c + "@col" + to_string(j)).second or
                    !seen.insert(c + "@box" + to_string(i/3) + to_string(j/3))
                        .second) return false;
            }

        return true;
    }
};
```

Lee × | Accepted ×

← All Submissions



Accepted 507 / 507 testcases passed

Itheng submitted at Nov 05, 2025 18:49

Editorial

Solution

Runtime

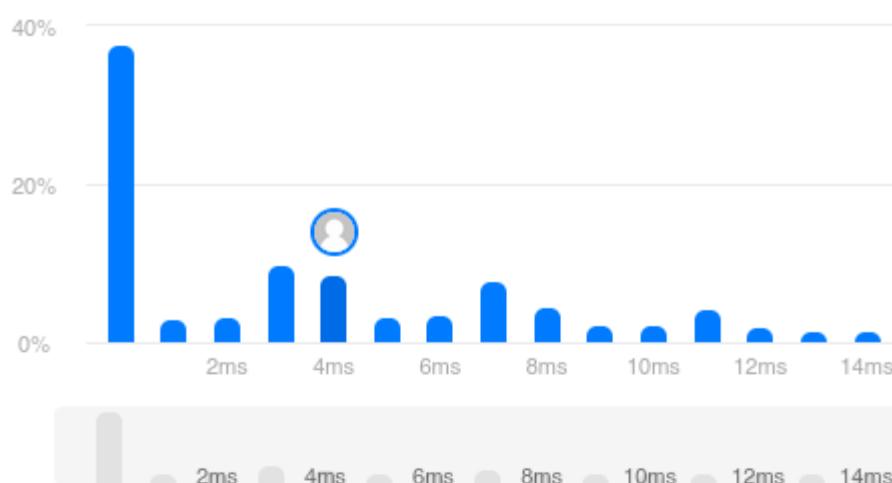


4 ms | Beats 46.08%

Analyze Complexity

Memory

25.35 MB | Beats 28.22%



Code | C++

```
class Solution {
public:
    bool isValidSudoku(vector < vector <char> >& board) {
        unordered_set <string> seen;

        for (int i = 0; i < 9; ++i)
            for (int j = 0; j < 9; ++j) {
                if (board[i][j] == '.')
```

▽ View more

## T6 Leetcode90 困难

```
class Solution {
public:
    vector < vector <int> > subsetsWithDup(vector <int>& nums) {
        vector < vector <int> > ans;
        ranges::sort(nums);
        dfs(nums, 0, {}, ans);
        return ans;
    }

private:
    void dfs(const vector <int>& nums, int s, vector <int>&& path,
             vector < vector <int> >& ans) {
        ans.push_back(path);

        for (int i = s; i < nums.size(); ++i) {
            if (i > s and nums[i] == nums[i - 1]) continue;
            path.push_back(nums[i]);
            dfs(nums, i + 1, std::move(path), ans);
            path.pop_back();
        }
    }
};
```

Lee × | Accepted ×

← All Submissions



Accepted 20 / 20 testcases passed

Itheng submitted at Nov 05, 2025 18:59

Editorial

Solution

⌚ Runtime

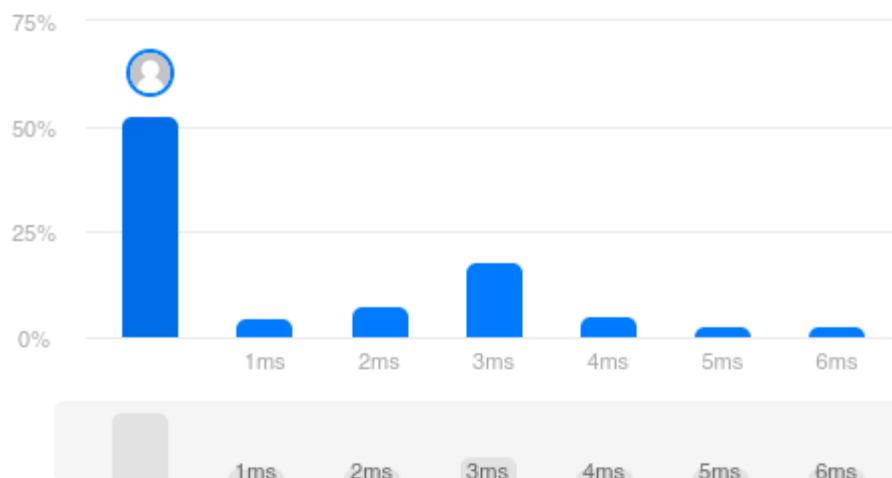


0 ms | Beats 100.00%

Analyze Complexity

Ⓜ Memory

10.49 MB | Beats 66.32%



Code | C++

```
class Solution {
public:
    vector<vector<int>> subsetsWithDup(vector<int>& nums) {
        vector<vector<int>> ans;
        ranges::sort(nums);
        dfs(nums, 0, {}, ans);
        return ans;
    }
}
```

View more