

Leetcode小組申請

申請日期：2020-09-16 (三)

QQ號：3068372624

破冰帖：<https://t.zsxq.com/E6QniYn>

1、編程能力：

大學是電機工程學系，大一學過兩門計算機導論(Java與C)，然後選修資料結構跟演算法，但當時這以理論跟紙本考試為主，沒有代碼實踐落地，但對於前端的HTML、CSS、javascript有一實際落地的經驗，研究所時重新以Duke Coursera的Java專項課重新刷過一遍，用的是Blue J編譯器，研究所最後論文是使用Java來開發SUMO交通模擬軟體，從訂單、貨物時間排程、軟體GUI展示等，但是，在今年2020年五月去面試時，遇到HackerRank題目實踐時，對於Hashtable：Ransom Note的解題很慌亂，甚至語法部分都不太熟，是種下面試失敗的原因，因此希望重藉由Leetcode的持續行動，把資料結構跟演算法的代碼落地實力積累下來，以因應面試。

2、小組管理期望

主要是維持小組刷題氛圍，維持行動的節奏跟執行小組規則，像是嚴格執行每日刷題任務，希望該小組刷題時能有整理或存放組員刷題紀錄，這些行動覆盤筆記中，希望能對題目做tag，像是已入群任務“two-sum”這題，那麼我在搜尋時，也能搜到他人對於相同題目的思路解法，以便能夠相互借鑒思考。

3、遇到困難的突破方法

首先，第一步要解決持續行動的問題，每天都要刷題，維持底線任務，從週一到週六，同常週六往往會覺得是週末所以刷題得弦沒有平日緊，因此週五就得提前在多刷幾題以當作存量，通常週六統整該週的刷題，總結覆盤，並切對於下週的題目做提前佈置。而對於，題型在解題遇到困難時，有兩種做法，

第一、先寫下自己的中文解題思路，可以Po上組內，詢問其他組員，但是不能確保一定會立刻收到回覆。

第二、上網Google，找尋github論壇中有的解法，以便參考。

目標：

現階是把空持續刷題的節奏，個人進階任務是每天2題，然後剛開始每週有2題medium1題Hard題，希望對於基礎的語法有先掌握，然後，慢慢地在把資料結構跟演算法慢慢實踐，同時也去對照HackerRank的題目，畢竟，我想面試的兩家公司都是用這個考，不過，殊途通歸，先把熟悉度建立起來才是最重要的事。

任務完成截圖：

The screenshot shows the LeetCode submission page for the 'Two Sum' problem. The page layout includes a navigation bar at the top with links to Explore, Problems, Mock, Contest, Discuss, and Store. The main content area is divided into three sections: a table of submission statistics, the problem description, and the Java code for the solution.

Time Submitted	Status	Runtime	Memory
09/16/2020 20:24	Accepted	84 ms	40.9 MB

```
1 class Solution {
2     public int[] twoSum(int[] nums, int target) {
3         int[] targetIndex = new int[2];
4         for(int i=0; i<nums.length; i++){
5             for(int j=i+1; j<nums.length; j++){
6                 if(nums[i]+nums[j]==target){
7                     targetIndex[0]=i;
8                     targetIndex[1]=j;
9                 }
10            }
11        }
12        //System.out.println("targetIndex:"+targetIndex);
13        return targetIndex;
14    }
15 }
```

Need more space? Try using full screen mode! [Got it!](#)

Your previous code was restored from your local storage. [Reset to default](#)

The screenshot shows the LeetCode problem page for 'Two Sum'. The page layout includes a navigation bar at the top with links to Explore, Problems, Mock, Contest, Discuss, and Store. The main content area is divided into three sections: the problem description, examples, and the Java code for the solution.

1. Two Sum

Easy 16835 607 Add to List Share

Given an array of integers `nums` and an integer `target`, return *indices of the two numbers such that they add up to `target`*.

You may assume that each input would have **exactly one solution**, and you may not use the same element twice.

You can return the answer in any order.

Example 1:

```
Input: nums = [2,7,11,15], target = 9
Output: [0,1]
Output: Because nums[0] + nums[1] == 9, we return [0, 1].
```

Example 2:

```
Input: nums = [3,2,4], target = 6
Output: [1,2]
```

```
1 class Solution {
2     public int[] twoSum(int[] nums, int target) {
3         int[] targetIndex = new int[2];
4         for(int i=0; i<nums.length; i++){
5             for(int j=i+1; j<nums.length; j++){
6                 if(nums[i]+nums[j]==target){
7                     targetIndex[0]=i;
8                     targetIndex[1]=j;
9                 }
10            }
11        }
12        //System.out.println("targetIndex:"+targetIndex);
13        return targetIndex;
14    }
15 }
```

Testcase Run Code Result Debugger

Accepted Runtime: 0 ms

Your input `[2,7,11,15]`
`9`

Output `[0,1]` [Diff](#)

Expected `[0,1]`

Console How to create a testcase [Run Code](#) [Submit](#)