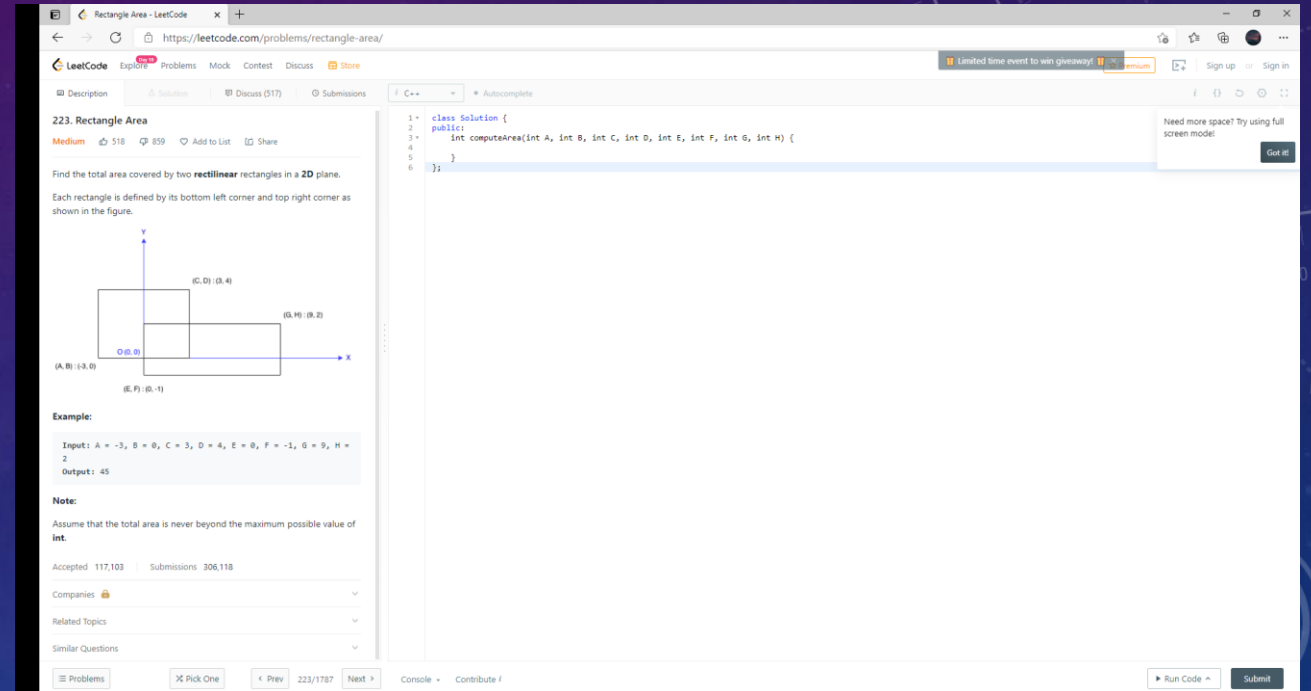
The background features a dark blue gradient with a starry space pattern. On the left side, there are several technical diagrams, including a large circular scale with numerical markings from 140 to 260, and various circular and curved lines with arrows indicating direction. The main text is centered on the right side.

CS3201(2020/2) TERM PROJECT

NGUYEN HA DUONG_6128301

PROBLEM INFO

- Problem Name: Rectangle Area
- Difficulty : Medium
- Link to the problem : [Rectangle Area - LeetCode](https://leetcode.com/problems/rectangle-area/)



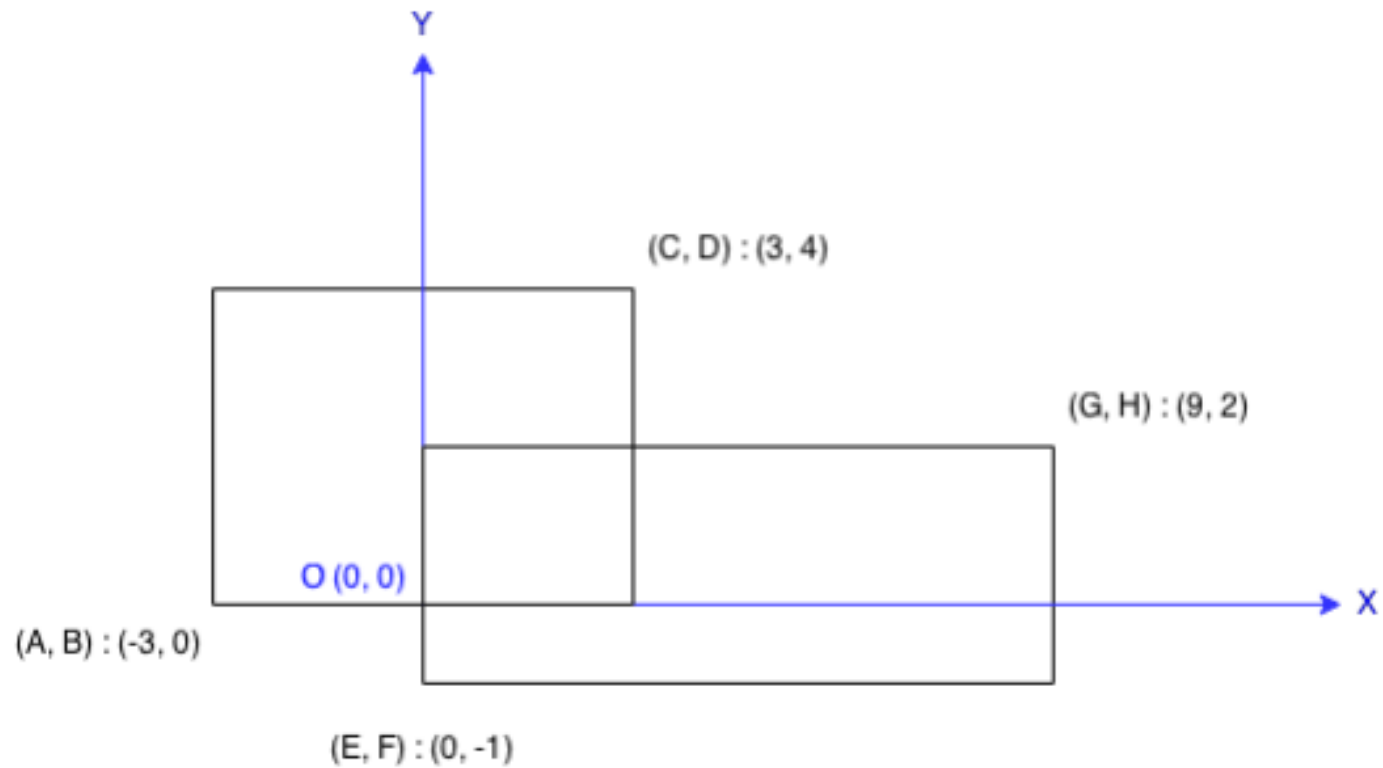
The screenshot shows the LeetCode problem page for "223. Rectangle Area". The problem is categorized as "Medium" and has 518 likes and 859 dislikes. The description asks to find the total area covered by two rectangular rectangles in a 2D plane, defined by their bottom-left and top-right corners. A diagram shows two overlapping rectangles with their corners labeled: (A, B) at (-3, 0), (E, F) at (-1, -1), (C, D) at (3, 4), and (G, H) at (4, 2). The example input is A = -3, B = 0, C = 3, D = 4, E = 0, F = -1, G = 4, H = 2, and the output is 45. The note states that the total area is never beyond the maximum possible value of int. The page also shows statistics: 117,103 accepted and 306,118 submissions. The code editor on the right shows a C++ solution template.

```
1 class Solution {
2 public:
3     int computeArea(int A, int B, int C, int D, int E, int F, int G, int H) {
4     }
5 }
6
```

PROBLEM INTRODUCTION

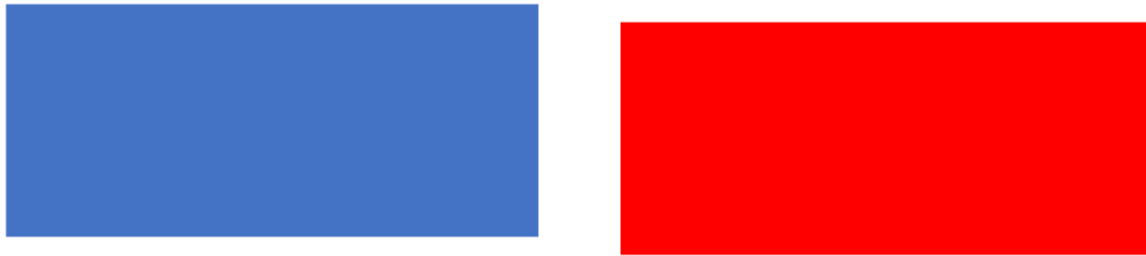
Find the total area covered by two **rectilinear** rectangles in a **2D** plane.

Each rectangle is defined by its bottom left corner and top right corner as shown in the figure.



PROBLEM ANALYSIS

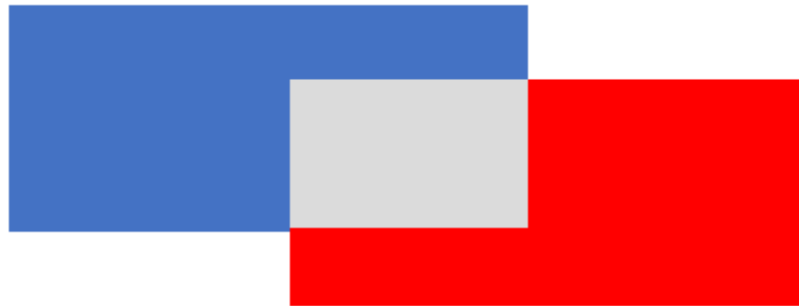
a, 2 rectangles do not overlap each other:



c, 1 rectangle in the other rectangle:



b, 2 rectangles overlap each other:

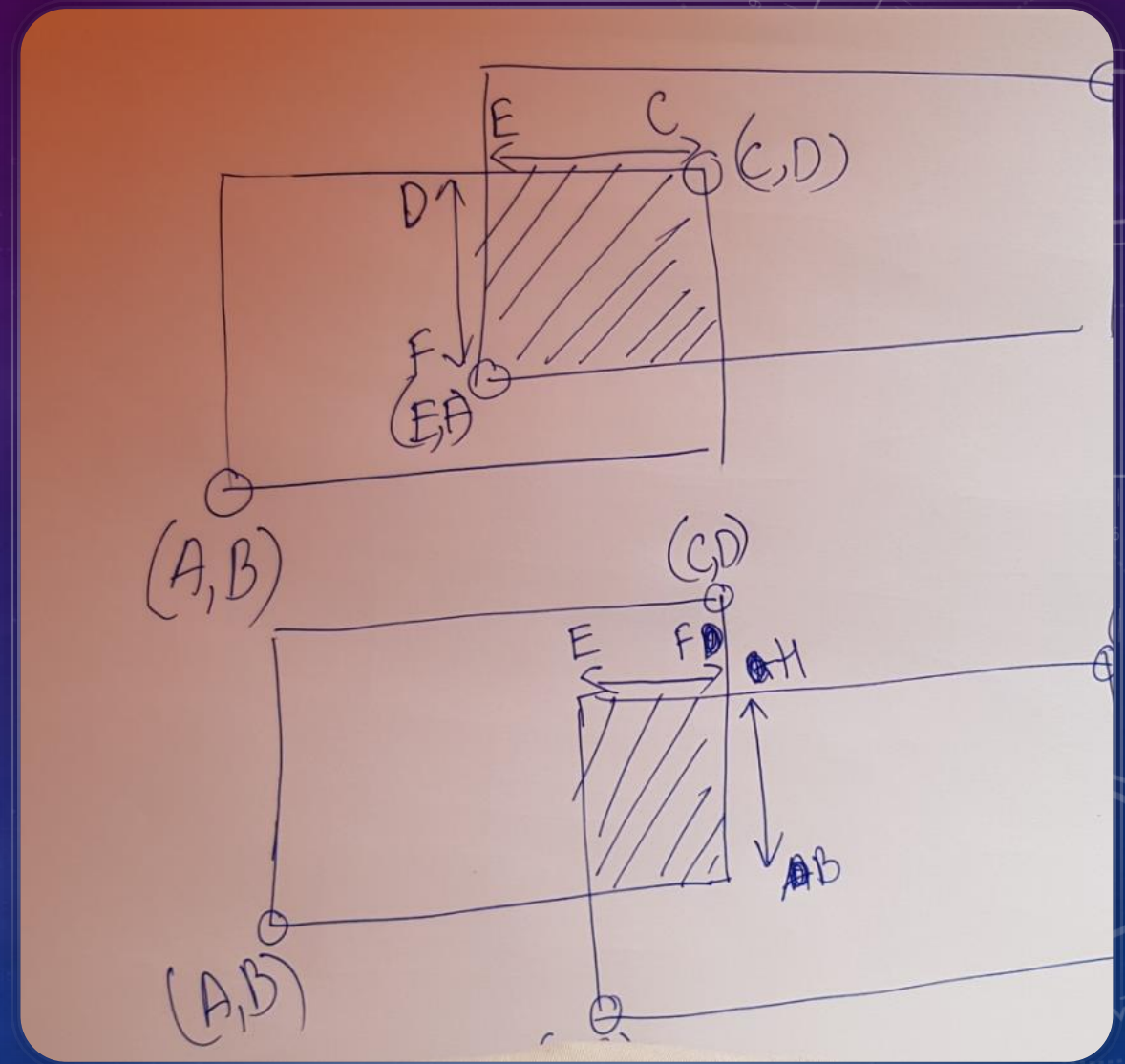


PROBLEM SOLUTION

- Divide into 2 cases: overlapped && not overlapped.
- If the problem is not overlapped: sum the Area of the 2 rectangles.
- Else : Sum up the Area of the 2 rectangles and then minus the Area of the overlapped part.

CALCULATE THE AREA OF THE OVERLAPPED PART

- In order to calculate the area of the overlapped part. We take the minimum value of x and y of 2 top-right and the max value of x and y of 2 bottom-left value.
- The new 2 point are the 2 conner of the overlapped area
- => Able to calculate the area of the overlapped part.



CODE_PYTHON

```
1 class Solution(object):
2     def computeArea(self, A, B, C, D, E, F, G, H):
3         S1 = abs(A-C) *abs(B-D)
4         S2 = abs(E-G) *abs(F-H)
5
6         if (A < G and B < H and E < C and F < D):
7             return S1 + S2 - (max(A,E) - min(C, G))*(max(B,F) - min(D,H))
8         return S1 + S2
9
10
```

(*) Additional:

```
class Solution(object):
    def computeArea(self, A, B, C, D, E, F, G, H):
        S1 = abs(A-C) *abs(B-D)
        S2 = abs(E-G) *abs(F-H)

        if ( E >= C or F >= D or G <= A or H <= B ):
            return (S1 + S2)
        return S1 + S2 - (max(A,E) - min(C, G))*(max(B,F) -
min(D,H))
```

SUBMISSION

Submission Detail

3082 / 3082 test cases passed.

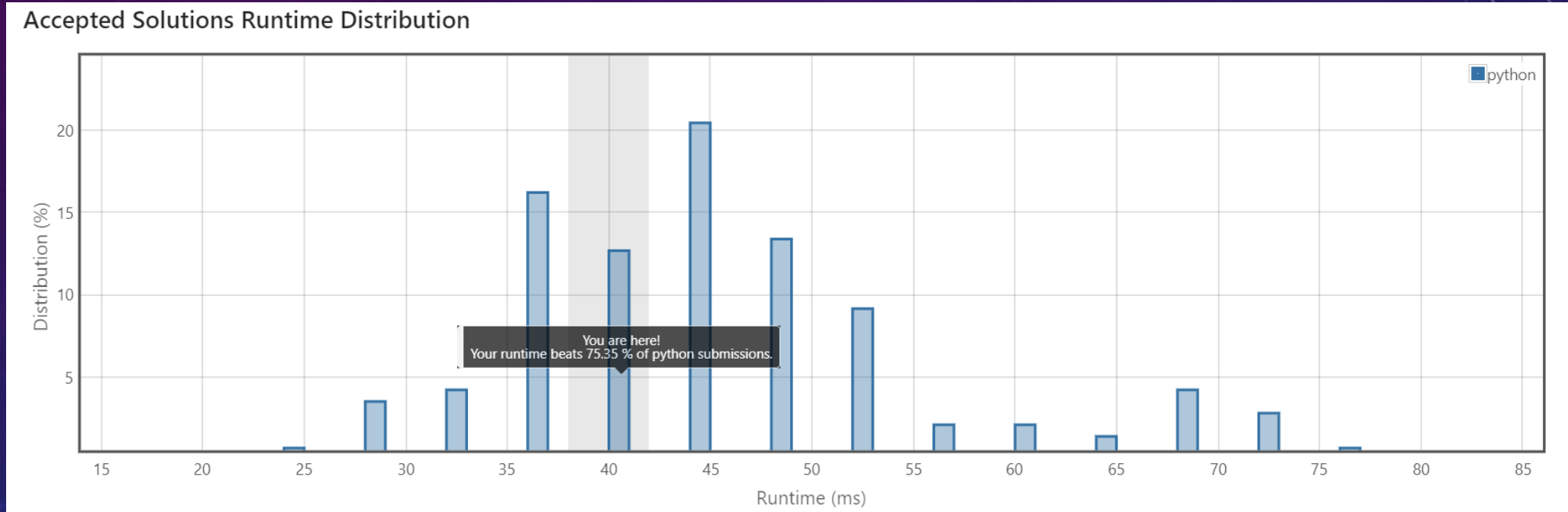
Runtime: **40 ms**

Memory Usage: **13.5 MB**

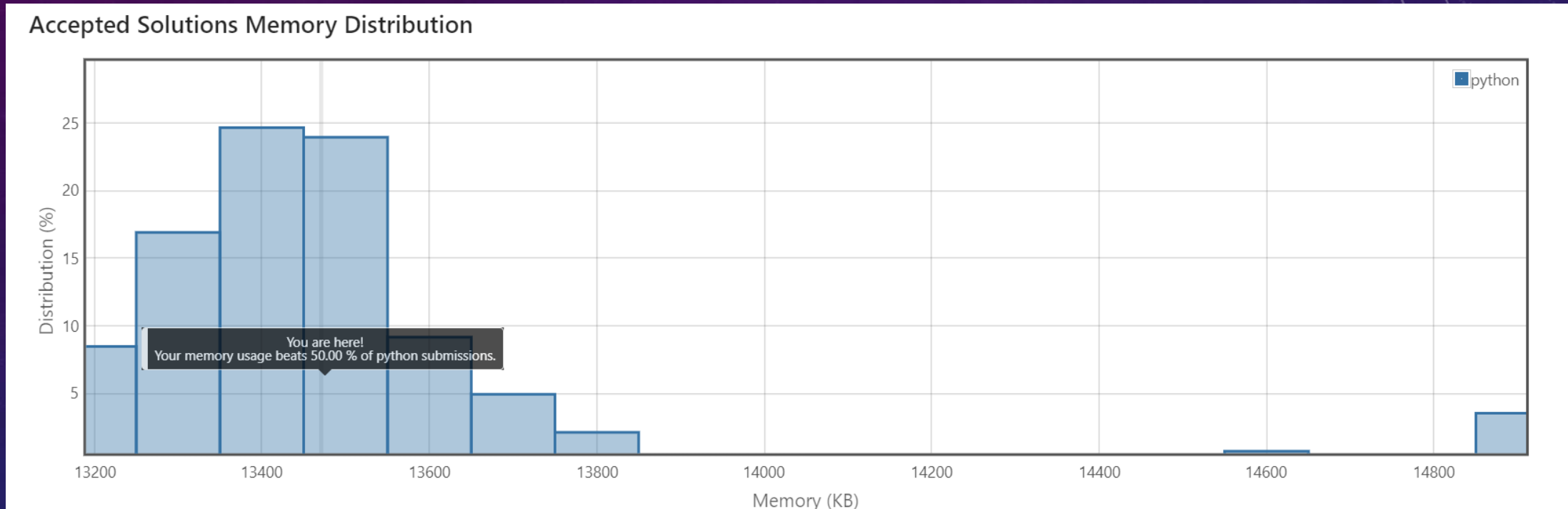
Status: **Accepted**

Submitted: **2 hours, 45 minutes ago**

RUNTIME AND MEMORY EVALUATION



RUNTIME AND MEMORY EVALUATION



Thank you!

3082 / 3082 test cases passed.

Runtime: 40 ms

Memory Usage: 13.5 MB

Status: **Accepted**

Submitted: 2 hours, 45 minutes ago

Addition: I forgot to put the picture of submission detail on my report.