

Problem 473: Matchsticks to Square

Problem Information

Difficulty: Medium

Acceptance Rate: 41.39%

Paid Only: No

Tags: Array, Dynamic Programming, Backtracking, Bit Manipulation, Bitmask

Problem Description

You are given an integer array `matchsticks` where `matchsticks[i]` is the length of the `i`th matchstick. You want to use **all the matchsticks** to make one square. You **should not break** any stick, but you can link them up, and each matchstick must be used **exactly one time**.

Return `true` if you can make this square and `false` otherwise.

Example 1:



Input: `matchsticks = [1,1,2,2,2]` **Output:** `true` **Explanation:** You can form a square with length 2, one side of the square came two sticks with length 1.

Example 2:

Input: `matchsticks = [3,3,3,3,4]` **Output:** `false` **Explanation:** You cannot find a way to form a square with all the matchsticks.

Constraints:

`1 <= matchsticks.length <= 15` `1 <= matchsticks[i] <= 108`

Code Snippets

C++:

```
class Solution {  
public:  
    bool makesquare(vector<int>& matchsticks) {  
  
    }  
};
```

Java:

```
class Solution {  
    public boolean makesquare(int[] matchsticks) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def makesquare(self, matchsticks: List[int]) -> bool:
```