# Problem: Can Construct Ransom Note

Return true if ransomNote can be constructed by using the letters from magazine.  
Each letter in magazine can be used **only once**.

**Input Example:**

ransomNote = "aabb"

magazine = "babaac"

# Step-by-Step Execution

## Step 1: Count characters in magazine

magazine = "babaac"

|  |  |  |
| --- | --- | --- |
| Index (ch - 'a') | Character | Count after loop |
| b → 1 | b | 1 |
| a → 0 | a | 1 |
| b → 1 | b | 2 |
| a → 0 | a | 2 |
| a → 0 | a | 3 |
| c → 2 | c | 1 |

✅ Final freq[] after magazine:

a → 3, b → 2, c → 1, rest → 0

## Step 2: Consume characters from ransomNote = "aabb"

|  |  |  |  |
| --- | --- | --- | --- |
| Char | freq Before | Action | freq After |
| a | 3 | use one → valid | 2 |
| a | 2 | use one → valid | 1 |
| b | 2 | use one → valid | 1 |
| b | 1 | use one → valid | 0 |

✅ All letters found and consumed → **return true**

# Example where it fails

ransomNote = "aabbc"

magazine = "aab"

|  |  |  |
| --- | --- | --- |
| Char | freq Before | Result |
| a | 2 | ok → freq = 1 |
| a | 1 | ok → freq = 0 |
| b | 1 | ok → freq = 0 |
| b | 0 | ❌ return false |

# Time and Space Complexity

|  |  |
| --- | --- |
| Metric | Value |
| Time | O(m + n) |
| Space | O(1) |

# Solution

public class Solution {

    public boolean canConstruct(String ransomNote, String magazine) {

        int[] freq = new int[26]; // Array for 'a' to 'z'

        // Count frequency of each letter in magazine

        for (char ch : magazine.toCharArray()) {

            freq[ch - 'a']++;

        }

        // Try to "consume" letters for ransomNote

        for (char ch : ransomNote.toCharArray()) {

            if (freq[ch - 'a'] == 0) {

                return false; // Letter not available

            }

            freq[ch - 'a']--; // Use the letter

        }

        return true; // All letters matched

    }

}