

6108. Decode the Message

My Submissions (/contest/weekly-contest-300/problems/decode-the-message/submissions/)

Back to Contest (/contest/weekly-contest-300/)

You are given the strings `key` and `message`, which represent a cipher key and a secret message, respectively. The steps to decode `message` are as follows:

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Easy

1. Use the **first** appearance of all 26 lowercase English letters in `key` as the **order** of the substitution table.

2. Align the substitution table with the regular English alphabet.

3. Each letter in `message` is then **substituted** using the table.

4. Spaces ' ' are transformed to themselves.
- For example, given `key = "happy boy"` (actual key would have **at least one** instance of each letter in the alphabet), we have the partial substitution table of ('h' -> 'a', 'a' -> 'b', 'p' -> 'c', 'y' -> 'd', 'b' -> 'e', 'o' -> 'f').

Return *the decoded message*.

Example 1:

t	h	e	q	u	i	c	k	b	r	o	w	n	f	x	j	m	p	s	v	l	a	z	y	d	g
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

Input: `key = "the quick brown fox jumps over the lazy dog"`, `message = "vkbs bs t suepuv"`
Output: `"this is a secret"`
Explanation: The diagram above shows the substitution table.
It is obtained by taking the first appearance of each letter in `"the quick brown fox jumps over the lazy dog"`.

Example 2:

e	l	j	u	x	h	p	w	n	y	r	d	g	t	q	k	v	i	s	z	c	f	m	a	b	o
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

Input: `key = "eljuxhpwnyrdgtqkviszcfmabo"`, `message = "zwx hnfx lqantp mnoeius ycgk vcnjrdb"`
Output: `"the five boxing wizards jump quickly"`
Explanation: The diagram above shows the substitution table.
It is obtained by taking the first appearance of each letter in `"eljuxhpwnyrdgtqkviszcfmabo"`.

Constraints:

- 26 <= `key.length` <= 2000
- `key` consists of lowercase English letters and ' '.
- `key` contains every letter in the English alphabet ('a' to 'z') **at least once**.
- 1 <= `message.length` <= 2000
- `message` consists of lowercase English letters and ' '.

JavaScript

1

2

3

4

5

6

7

8


```
const ord = (c) => c.charCodeAt();
const char = (ascii) => String.fromCharCode(ascii);
const decodeMessage = (key, message) => {
  let m = new Map(), start = 97, used = new Set(), res = '';
  for (const c of key) {
    if (c == ' ' || used.has(c)) continue;
    m.set(c, char(start++));
  }
}
```

```
9      used.add(c);
10     }
11     for (const c of message) res += c == ' ' ? c : m.get(c);
12     return res;
13 };
```

☐ Custom Testcase[Use Example Testcases](#)[Run](#)[Submit](#)**Submission Result: Accepted** (</submissions/detail/737052626/>) [More Details >](/submissions/detail/737052626/)

Share your acceptance!

Copyright © 2022 LeetCode

[Help Center \(/support/\)](/support/) | [Jobs \(/jobs/\)](/jobs/) | [Bug Bounty \(/bugbounty/\)](/bugbounty/) | [Online Interview \(/interview/\)](/interview/) | [Students \(/student/\)](/student/) | [Terms \(/terms/\)](/terms/) | [Privacy Policy \(/privacy/\)](/privacy/) [United States \(/region/\)](/region/)