

## 100126. Number of Strings Which Can Be Rearranged to Contain Substring

[My Submissions \(/contest/biweekly-contest-117/problems/number-of-strings-which-can-be-rearranged-to-contain-substring/submissions/\)](#)

[Back to Contest \(/contest/biweekly-contest-117/\)](#)

You are given an integer  $n$ .

A string  $s$  is called **good** if it contains only lowercase English characters **and** it is possible to rearrange the characters of  $s$  such that the new string contains "leet" as a **substring**.

For example:

- The string "lteer" is good because we can rearrange it to form "leetr".
- "letl" is not good because we cannot rearrange it to contain "leet" as a substring.

Return the **total** number of good strings of length  $n$ .

Since the answer may be large, return it **modulo**  $10^9 + 7$ .

A **substring** is a contiguous sequence of characters within a string.

User Accepted:	95
User Tried:	194
Total Accepted:	95
Total Submissions:	305
Difficulty:	Medium

### Example 1:

**Input:**  $n = 4$

**Output:** 12

**Explanation:** The 12 strings which can be rearranged to have "leet" as a substring are: "eelt", "eetl", "elet", "elte"

### Example 2:

**Input:**  $n = 10$

**Output:** 83943898

**Explanation:** The number of strings with length 10 which can be rearranged to have "leet" as a substring is 5260839475

### Constraints:

- $1 \leq n \leq 10^5$

JavaScript


```
1 const powmod = (a, b, mod) => { let r = 1; while (b > 0) { if (b & 1) r = multi_mod(r, a, mod); b >>= 1; a = multi_mod(a, a, mod); } return r; };
2 const multi_mod = (x, y, mod) => Number((ll(x) * ll(y) % ll(mod)));
3 const minus_mod = (x, y, mod) => ((x - y) % mod + mod) % mod;
4
5 const ll = BigInt, mod = 1e9 + 7;
6 const stringCount = (n) => {
7   let tot = powmod(26, n, mod),
8       l = t = powmod(25, n, mod),
9       e = n * powmod(25, n - 1, mod) + powmod(25, n, mod),
10      lt = powmod(24, n, mod),
11      le = et = n * powmod(24, n - 1, mod) + powmod(24, n, mod),
12      LET = n * powmod(23, n - 1, mod) + powmod(23, n, mod)
13   let res = tot;
14   res = minus_mod(res, l, mod);
15   res = minus_mod(res, t, mod);
16   res = minus_mod(res, e, mod);
17   res += lt;
18   res %= mod;
```

```
19     res += le;  
20     res %= mod;  
21     res += et;  
22     res %= mod;  
23     res = minus_mod(res, LET, mod);  
24     return res;  
25 };
```

☐ Custom Testcase[Use Example Testcases](#)[Run](#)[Submit](#)**Submission Result: Accepted** (</submissions/detail/1096708922/>) ⓘ[More Details > \(/submissions/detail/1096708922/\)](/submissions/detail/1096708922/)

Share your acceptance!

Copyright © 2023 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/)