Given 3 strings of only lowercase letter you have to count the number of ways you can construct the third string by combining two subsequences from the first two strings.

After deleting 0 or more characters from a string we can get its subsequence. For example “a”, “b”, “c”, “ab”, “ac”, “bc” and “abc” all the strings are the subsequences of “abc”. A subsequence may also be empty.

Now suppose there are two subsequences “abc” and “de”. By combining them you can get the following strings “abcde”, “abdce”, “abdec”, “adbce”, “adbec”, “adebc”, “dabce”, “dabec”, “daebc” and “deabc”.

Input

The first line of the input contains a single integer T $(0 < T < 271)$ indicating the number of test cases. Each test case contains 3 strings containing only lowercase characters. The lengths of the strings are between 1 and 60.

Output

For each test case output a single integer denoting the number of ways you can construct the third string from the first two string by the above way. The result may be very large. You should output the $result\%10007$.

Sample Input

```
2
abc abc abc
abbcd bccde abcde
```

Sample Output

```
8
18
```