

Alphabet Soup

Filename: alphabet

The Problem

You are creating mugs with (two) initials for sale. Your limiting factor is the number of peel off letters you have to use. Of course, since you want to maximize your profit, you want to make as many mugs as possible. The problem is that your sense of style prevents you from using any pair of initials. First off, all of your peel off letters come in either red or green, but your sense of style dictates that the first initial must be red while the second must be green. Secondly, you don't like initials that are too close together in the alphabet. Specifically, you disallow a pair of letters if the difference of their ordinal value is less than 3. (For example, AD and DA are allowed, but ZX and XZ are not.) Given a list of the red letters you have and the green letters you have, determine the maximum number of mugs you can make, on account of the letters.

The Input

The first line of the input file will contain a single positive integer, n ($n \leq 100$), representing the number of input cases. The input cases follow, one per line. Each input case will have two space separated strings of lowercase letters, each with a length in between 1 and 50 characters, inclusive. The first string represents each red peel off letter while the second string represents each green peel off letter.

The Output

For each input case, output the maximum number of mugs that can be made, based on the letters you have.

Sample Input

```
3
abc abc
abcde wxyz
abcde abcde
```

Sample Output

```
0
4
4
```