ACM-ICPC Live Archive

2812 UNISON

A union is an instance of playing the same pitch or of playing in octave. We want to arrange new polyphonic music with three different monophonic music scores of arbitrary length. Each monophonic music score has only 7 tones, C(do), D(re), E(mi), F(fa), G(sol), A(la), B(si) with the same interval.

You can see polyphonic music X with three monophonic music scores (Music1, Music2, Music3) in Figure 1. Note that the lengths of Music1, Music2, and Music3 are 12, 12, 11, respectively. '&' is the symbol of a pause and means that no tone is played in that position. Music X is the same as playing 3 different monophonic music scores (Music1, 2, 3) simultaneously. The tones in each column of Figure-1 denote a set of tones to be played simultaneously in polyphonic music.

	Start											end
Playing column	1	2	3	4	5	**6**	7	8	9	**10**	11	**12**
Music1	A	C	A	G	E	B	D	A	D	D	A	A
Music2	F	C	G	A	G	B	B	D	A	D	F	A
Music3	A	F	C	A	B	B	D	A	D	D	F	&
Unison weight	0	0	0	0	0	3	0	0	0	3	0	1

Figure 1: A newly arranged polyphonic music X with unison weight 7.

Let us define "unison" in this problem. A unison is an instance where the same tones are played simultaneously in a score. Thus, column1 = {A, F, A} and column2 = {C, C, F} are not unisons, so the music score X has only three unisons: column6={B, B, B}, column10 = {D, D, D}, and column12 = {A, A, &} . Formally, a column $M = \{x, y, z\}$ is a unison, if M has three of the same tones or two of the same tones plus '&'. If $x = y = z \neq$ &, then the weight of the unison is 3. If M has two of the same tones and a single '&', then the weight of the column is 1. The sum of all unisons in a score is called the total unison weight. The following table shows the unison weight for each tone set.

Tones in a column	**Unison weight**
C C C	3
D D D	3
E E E	3
.....	3
B B B	3
C C &	1
D D &	1
.....	1
B B &	1
All others	0

Table 1: Unison weight table for the tones in a column.

The unison weight of score X in Figure-1 is 3 + 3 + 1 = 7. However, if we arrange X to get another music by inserting pause symbols smartly, then we can increase the total unison weight significantly. Figure 2 shows another arrangement Y with unison weight 25.

	Start															end
Playing column	**1**	**2**	**3**	4	**5**	**6**	**7**	8	**9**	**10**	**11**	**12**	**13**	14	**15**	**16**
Music1	A	&	C	&	A	G	&	E	B	D	A	D	D	A	&	A
Music2	&	F	C	G	A	G	B	&	B	D	A	&	D	&	F	A
Music3	A	F	C	&	A	&	B	&	B	D	A	D	D	&	F	&
Unison weight	1	1	3	0	3	1	1	0	3	3	3	1	3	0	1	1

Figure 2: Another polyphonic music Y with unison weight 25.

You are asked to compute the maximal unison weight from three different music scores by optimally inserting '&' symbols in each score. **One important constraint on inserting '&' is that you should not insert more than one consecutive '&' in a music score.** See Figure 3 for this constraint. Also, you are not allowed to modify the tones (melody) or to change the sequence of tones. Keep in mind that your arrangement can only insert &'s separately to get a maximal unison.

Write a program that finds the maximal unison weight by using an optimal arrangement.

Playing	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Music1	&	G	&	A	&	G	B	&	C	F	A	&	C	B
Music2	A	G	E	&	&	&	B	G	&	&	A	G	C	&
Music3	&	G	E	A	C	D	&	E	C	F	A	G	C	&

Figure 3: This arrangement is not allowed (infeasible arrangement) because Music2 includes more than one &.

Input

The input consists of T test cases. The number of test cases (T) is given on the first line of the input file. Each test case starts with three lines containing a string of monophonic music scores consisting of a symbol set ={`C`,`D`,`E`,`F`,`G`,`A`,`B`}. **There is a blank line between test cases.** The maximal length of each monophonic music score string is 100.

Output

For each test case, your program reports only one number, the maximal total unison weight of each testing case in a line. If there is no arrangement satisfying the constraint, then print '`-1`' as the output. The following shows sample input and output for four test cases.

Sample Input

```
4
ABCDEF
BCDEFG
DEFGAB

GABBCDEACF
GAFGFCBBDEDD
AFAFGCEECDEDF

BCDEFGGADEF
BCDE
CDEFGABBA

GAGABDEDCGAFFAB
```

```
DAFFAGAEECCBA
GCBDAAEGEAGAFF
```

Sample Output

```
5
15
-1
14
```