

Kerbside recycling has come to New Zealand, and every city from Auckland to Invercargill has leapt on to the band wagon. The bins come in 5 different colours — red, orange, yellow, green and blue — and 5 wastes have been identified for recycling — Plastic, Glass, Aluminium, Steel, and Newspaper. Obviously there has been no coordination between cities, so each city has allocated wastes to bins in an arbitrary fashion. Now that the government has solved the minor problems of today (such as reorganising Health, Welfare and Education), they are looking around for further challenges. The Minister for Environmental Doodads wishes to introduce the “Regularisation of Allocation of Solid Waste to Bin Colour Bill” to Parliament, but in order to do so needs to determine an allocation of his own. Being a firm believer in democracy (well some of the time anyway), he surveys all the cities that are using this recycling method. From these data he wishes to determine the city whose allocation scheme (if imposed on the rest of the country) would cause the least impact, that is would cause the smallest number of changes in the allocations of the other cities. Note that the sizes of the cities is not an issue, after all this is a democracy with the slogan “One City, One Vote”.

Write a program that will read in a series of allocations of wastes to bins and determine which city’s allocation scheme should be chosen. Note that there will always be a clear winner.

Input

Input will consist of a series of blocks. Each block will consist of a series of lines and each line will contain a series of allocations in the form shown in the example. There may be up to 100 cities in a block. Each block will be terminated by a line starting with ‘e’. The entire file will be terminated by a line consisting of a single ‘#’.

Output

Output will consist of a series of lines, one for each block in the input. Each line will consist of the number of the city that should be adopted as a national example.

Sample Input

```
r/P,o/G,y/S,g/A,b/N
r/G,o/P,y/S,g/A,b/N
r/P,y/S,o/G,g/N,b/A
r/P,o/S,y/A,g/G,b/N
e
r/G,o/P,y/S,g/A,b/N
r/P,y/S,o/G,g/N,b/A
r/P,o/S,y/A,g/G,b/N
r/P,o/G,y/S,g/A,b/N
ecclesiastical
#
```

Sample Output

```
1
4
```