



Limite de mémoire: 256 MB

unes des autres, en d'autres termes, pour chaque requête, on commence par la même configuration initiale décrite dans la première partie de l'énoncé.

Entrées

1. Un entier n ($1 \leq n \leq 10^5$): le nombre de disques.
 2. Une grille de $10 \times n$ entiers positifs, où $a_{j,i} \leq 10^9$ ($0 \leq j \leq 9, 1 \leq i \leq n$) spécifie le nombre de fois où le chiffre j apparaît sur le disque d_i . Notez que chaque disque contient un nombre pair de chiffres.
 3. Un entier q ($1 \leq q \leq 2 \times 10^6$): le nombre de codes.
 4. q codes x_1, x_2, \dots, x_q , chacun de longueur $2n$: les codes cibles.
- Il est certain que $nq \leq 7 \times 10^6$

Sorties

Pour chaque code cible x_k ($1 \leq k \leq q$), afficher:

- Sur la première ligne, afficher POSSIBLE s'il est possible d'aligner les disques pour obtenir le code x_k .
- Sur la deuxième ligne, le **nombre minimum de rotations** nécessaires afin d'obtenir le code x_k , si c'est impossible afficher -1.

Si la première ligne de sortie est correcte tandis que la deuxième ne l'est pas, la moitié de la note est donnée.

Contraintes

- $1 \leq n \leq 10^5$
- $0 \leq a_{j,i} \leq 10^9$
- $1 \leq q \leq 2 \times 10^6$
- $nq \leq 7 \times 10^6$

Sous-tâches

Sous-tâche	Score	Contraintes
1	6	$n = q = 1$
2	6	Chaque disque contient un unique chiffre
3	10	Pour tout i , $\sum_{j=0}^9 a_{j,i} \leq 2$
4	8	$n = 1$
5	20	Pour tout $i, j, x, y : a_{i,j} = a_{x,y}$
6	30	$nq \leq 10^5$
7	20	Aucune contrainte supplémentaire

Exemples

Entrée

```
4
0 3 1 2 0 0 0 0 0 0
0 0 0 0 4 2 3 0 1 0
0 2 0 0 4 1 3 1 0 1
0 2 0 0 4 1 3 1 0 5
4
65521411
65521419
75521414
75521410
```

Note : Ceci est l'exemple dessiné sur la page 1.

Sortie

```
POSSIBLE
0
POSSIBLE
1
POSSIBLE
2
IMPOSSIBLE
-1
```

Entrée

```
1
4 3 2 3 1 1 4 3 9 2
1
71
```

Sortie

```
POSSIBLE
4
```