

K

8-ball Rack

Input: Standard Input
 Output: Standard Output

8-ball is a pool game typically played with **15** object balls. The object balls are of following types:

1. Solids, numbered from **1-7**
2. Stripes, numbered from **9-15**
3. Eight ball: as the name suggests, numbered **8** and is a black solid ball.



Figure 1: A valid 8 ball rack

In a valid 8-ball rack, the object balls are arranged in a triangle as the figure above, following these rules:

1. The eight ball has to be in the center.
2. The two rear corner balls have to be of different types. If one of them is stripes, then the other one must be from solids. In the picture above, two rear corner balls are 11 and 5. **11** is a stripe and **5** is a solid.

You are given an **8** ball rack (possibly incomplete). If a ball is not present in the rack, then you can assume that they are outside of the rack. You have choice of three different operations:

1. Swap any two balls already in the rack.

2. Pick a ball that's not already in the rack from outside and put that into the rack.
 3. Pick a ball that's already in the rack and put it outside.
- Given the initial rack and the cost of the different types of operations, find the minimum cost to form a valid 8-ball rack.

Input

The input file starts with a positive integer **T** ($T \leq 500$) denoting the number of test cases. Each of the **T** test cases starts with three non negative integers, **S**, **P**, and **R**, where,

1. **S** denotes the cost of swapping positions of any two balls.
2. **P** denotes the cost of adding a ball to the rack from outside.
3. **R** denotes the cost of removing a ball from the rack. Note that you can (well, you will have to) return this ball to any place in the rack later.
4. $0 \leq S, P, R \leq 100000$

Then come exactly five lines denoting the rack's formation. The *i*'th line has *i* non negative numbers in it. If the *j*'th number on the *i*'th line is **0**, then it means that position is empty. It is guaranteed that no non-zero number appears more than once in the rack.

For more on input specification, please see the sample input section.

Output

For each case, print a single line, "**Case #x: y**", where *x* is the case number and *y* an integer number which denotes the minimum cost to form a valid rack for that case.

Sample Input

```
2
1 1 1
9
7 12
15 8 1
6 10 3 14
11 2 13 4 5
1 2 3
9
7 12
15 8 1
6 10 3 14
0 0 0 0 0
```

Output for Sample Input

```
Case 1: 0
Case 2: 10
```

Note:

Case 1 is the rack given in the figure 1, which is already valid. So the output is 0

Case 2 is the rack given in the figure 1 only with the difference that, all the balls in the last row is missing. A valid rack can be formed with exactly 5 type 2 operations. As $P = 2$, the output for case 2 is $2 * 5 = 10$.