code jam

printf("hello, world!\n");

Problem F. **Teleporters**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the **Quick-Start Guide** to get started.

| Small input 10 points | Solve F-small |
| Large input 30 points | Solve F-large |

Problem

A short, short time into the future, in a nearby galaxy, you find yourself wanting to take a little trip and get away from the responsibilities of being Planet Thundera's only manufacturer of yarn. You decide to travel to Planet Care-a-Lot, the most relaxing planet there is. To travel, you are going to use the network of interstellar teleporters.

A teleporter is a small machine floating around somewhere in space. You can use it remotely from any point in space, but, due to the conservation of teleportation distance principle, it can teleport you to any other point in space at exactly the same L1 distance from the teleporter as your L1 distance to it before the teleportation. The L1 distance between two points at coordinates (x_0, y_0, z_0) and (x_1, y_1, z_1) is given by $|x_0 - x_1| + |y_0 - y_1| + |z_0 - z_1|$. Unfortunately, your space jetpack is broken, so you cannot move around on your own; to travel, you can only use the teleporters. You start at Planet Thundera. You can use a teleporter to travel from Planet Thundera to a point p_1, then use another to get from p_1 to p_2, and so on. The last teleportation must take you exactly to Planet Care-a-Lot.

Given the locations in 3-dimensional space of both planets and all the available teleporters, find out if it is possible for you to make the trip using only teleporters. If the trip can be made, what is the minimum number of teleportations needed to get to your destination? (Even if two teleportations use the same teleporter, they still count as separate teleportations.)

The input is given as points with coordinates that are all integers that fall within a certain range. However, you are allowed to teleport to intermediate points with integer or non-integer coordinates, and there are no range restrictions on the points you can visit.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case starts with a single line with a single integer **N**, the number of teleporters available. Then, **N**+2 lines follow, each containing three integers X_i, Y_i, and Z_i. The first of these lines represents the coordinates of your home planet, Thundera. The second of these lines represents the coordinates of your destination planet, Care-A-Lot. Each of the remaining **N** lines represents the coordinates of one of the teleporters.

Output

For each test case, output one line containing `Case #x: y`, where x is the test case number (starting from 1) and y is `IMPOSSIBLE` if it is not possible to get from Thundera to Care-A-Lot using only the available teleporters, or, if it is possible, an integer representing the minimum number of teleportations needed.

Limits

$1 \le T \le 100$.
$(X_i, Y_i, Z_i) \neq (X_j, Y_j, Z_j)$ for all i ≠ j. (No two described objects have the same coordinates.)

Small dataset

$1 \le N \le 100$.
$-10^3 \le X_i \le 10^3$, for all i.
$-10^3 \le Y_i \le 10^3$, for all i.
$-10^3 \le Z_i \le 10^3$, for all i.

Large dataset

$1 \le N \le 150$.
$-10^{12} \le X_i \le 10^{12}$, for all i.
$-10^{12} \le Y_i \le 10^{12}$, for all i.
$-10^{12} \le Z_i \le 10^{12}$, for all i.

Sample

```
Input       Output

3           Case #1: IMPOSSIBLE
1           Case #2: 3
0 0 0       Case #3: 2
0 4 0
0 3 0
2
0 0 1
0 0 11
0 0 3
0 0 0
3
0 0 0
6 2 0
6 0 0
3 0 0
6 1 0
```

In Sample Case #1, the only teleporter is exactly 3 units away from Thundera, and we can only use it to go to another position that is *exactly* 3 units away from the teleporter. From that position, we can still only reach other positions that are exactly 3 units away from the teleporter. Since Care-a-Lot is 1 unit away from the teleporter, we can never reach it.

In Sample Case #2, the optimal strategy is to first use the teleporter at (0, 0, 3) to travel to (0, 0, 5). Then, from there, use the teleporter at (0, 0, 0) to travel to (0, 0, -5). Finally, from there, use the teleporter at (0, 0, 3) again to travel to (0, 0, 11). Note that the two uses of the teleporter at (0, 0, 3) cause us to travel different distances, because we are at different distances from the teleporter each time. Also note that the two uses of that teleporter count as two separate teleportations.

In Sample Case #3, the optimal strategy is to first use the teleporter at (3, 0, 0) to travel to (6, 0, 0). Then, from there, use the teleporter at (6, 1, 0) to travel to (6, 2, 0). Note that even though there was a teleporter at (6, 0, 0), merely occupying the same point as a teleporter does not count as using it.

Powered by

Google Cloud Platform