

Spinning Blade

Problem

Being bored with the traps in your secret hideout design, you decided to go for something classical, but always enjoyable - the *spinning blade*. You ordered a really heavy metal sheet out of which you will cut the blade; a uniform square **C**-by-**R** grid will be painted on the sheet. You have determined the best shape for the blade -- you will first cut a large square consisting of **K**-by-**K** grid cells, where $\mathbf{K} \geq 3$. Then, you will cut out the four 1-by-1 corner cells out of the square to end up with a *blade*. After determining all this, you started waiting for the sheet to arrive.

When the sheet arrived, you were shocked to find out that the sheet had imperfections in it! You expected each cell to have mass **D**, but it turned out that the mass can vary a bit because of differences in thickness. This is bad because you want to insert a shaft exactly in the center of the blade and spin it very fast, so the center of mass of the blade must be exactly in its center as well. The definition of the center of mass of a flat body can be found below.

Given the grid and the mass of each cell, what is the largest possible size of the blade you can make so that the center of mass is exactly in its center?

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each one starts with a line containing 3 integers: **R**, **C** and **D** — the dimensions of the grid and the mass you expected each cell to have. The next **R** lines each contain **C** digits w_{ij} each, giving the differences between the actual and the expected mass of the grid cells. Each cell has a uniform density, but could have an integer mass between $\mathbf{D} + 0$ and $\mathbf{D} + 9$, inclusive.

Output

For each test case, output one line containing "Case #x: **K**", where x is the case number (starting from 1) and **K** is the largest possible size of the blade you can cut out. If no acceptable blade of size at least 3 can be found, print "IMPOSSIBLE" instead.

Limits

$$1 \leq \mathbf{T} \leq 20.$$

$$0 \leq w_{ij} \leq 9.$$

The size of the input file will not exceed 625KB.

Memory limit: 1GB.

Small dataset (Test set 1 - Visible)

$$3 \leq \mathbf{R} \leq 10.$$

$$3 \leq \mathbf{C} \leq 10.$$

$$1 \leq \mathbf{D} \leq 100.$$

Time limit: 30 seconds.

Large dataset (Test set 2 - Hidden)

$$3 \leq \mathbf{R} \leq 500.$$

$$3 \leq \mathbf{C} \leq 500.$$

$1 \leq D \leq 10^6$.

Time limit: 60 seconds.

Sample

Sample Input

```
2
6 7 2
1111111
1122271
1211521
1329131
1242121
1122211
3 3 7
123
234
345
```

Sample Output

```
Case #1: 5
Case #2: IMPOSSIBLE
```

Note

The *center of mass* of a 2D object is formally defined as a point \mathbf{c} . If you compute the sum of $(\mathbf{p} - \mathbf{c}) * \text{mass}(\mathbf{p})$ for all points \mathbf{p} in the object, you must get $\mathbf{0}$. Here, \mathbf{p} , \mathbf{c} and $\mathbf{0}$ are two-dimensional vectors. This definition also works if you treat each grid cell as a "point", with all of its mass at its center.

In real life, you could place your finger under a flat object's center of mass, and balance that object on your finger. It would not fall.

To illustrate with an example, the only blade that is possible to cut out in the second sample test case, the 3x3 blade created by cutting away the corners, has its center of mass at the point (1.54, 1.46), where we assume the bottom-left corner of the sheet has coordinates (0, 0), and the coordinates grow right and up, respectively. This is verified by checking the following equality: $(-1.04, 0.04) * 9 + (-0.04, 1.04) * 9 + (-0.04, 0.04) * 10 + (-0.04, -0.96) * 11 + (0.96, 0.04) * 11 = (0, 0)$.