



Problem F (Program filename: F.cpp, F.dpr, or F.java)

Points

Let p_1, p_2, \dots, p_n be n points on the plane. We have m rules of form $p_i \text{ rel } p_j$, each inform us that the relation rel holds among the locations of points p_i and p_j on the plane. For example, " $p_i \text{ NE } p_j$ " indicates that point p_j is located NorthEast of point p_i . There are eight different relations $\{N, E, S, W, NE, NW, SE, SW\}$, corresponding to the eight directions on the plane. Let (x_i, y_i) and (x_j, y_j) be the coordinates of p_i , and p_j respectively. Then $p_i \text{ rel } p_j$ exactly means one of the following, depending on the value of rel :

1. N stands for North. This means that $x_j = x_i$ and $y_j > y_i$,
2. E stands for East. This means that $x_j > x_i$ and $y_j = y_i$,
3. S stands for South. This means that $x_j = x_i$ and $y_j < y_i$,
4. W stands for West. This means that $x_j < x_i$ and $y_j = y_i$,
5. NE stands for NorthEast. This means that $x_j > x_i$ and $y_j > y_i$,
6. NW stands for NorthWest. This means that $x_j < x_i$ and $y_j > y_i$,
7. SE stands for SouthEast. This means that $x_j > x_i$ and $y_j < y_i$, and
8. SW stands for SouthWest. This means that $x_j < x_i$ and $y_j < y_i$.

The problem is to determine whether it possible to locate p_1, p_2, \dots, p_n on the plane so that all given rules are satisfied.

Input (filename: F.in)

The first line of the input contains a single integer t ($1 \leq t \leq 20$) which is the number of test cases in the input. The first line of each test case contains two integers n ($2 \leq n \leq 500$) which is the number of points and m ($1 \leq m \leq 10^4$) which is the number of rules. In each of the following m lines, there is one rule of the form $i \text{ rel } j$ which means that p_i has relation rel with p_j .

Output (filename: F.out)

The output contains one line per each test case containing one of the words POSSIBLE or IMPOSSIBLE indicating if the set of points in the test case can be located on the plane according to the given rules.

Sample Input

```
2
3 2
1 N 2
2 N 1
6 6
1 E 2
1 E 3
2 N 4
3 NW 5
```

4 SW 6
6 NE 5

Sample Output

IMPOSSIBLE
POSSIBLE