

Cave Exploration

A long time ago one man said that he had explored the corridors of one cave. This means that he was in all corridors of the cave. Corridors are really horizontal or vertical segments. A corridor is treated as visited if he was in at least one point of the corridor.

Now you want to know if this is true. You have a map of the cave, and you know that the explorer used the following algorithm: he turns left if he can, if he can't he goes straight, if he can't he turns right, if he can't he turns back. Exploration ends when the man reaches the entry point for the second time. Your task is to count how many corridors weren't visited by explorer.

Input

In the first line there is an integer **T** ($T \leq 20$) - the number of different maps. For each map in the first line there is an integer **N** ($N \leq 1000$) - the number of corridors. It is known that no two vertical corridors have a common point and no two horizontal corridors have a common point. The next **N** lines contain the following information: the line starts with one of the characters 'V' or 'H' - vertical or horizontal corridor. Then one Y-coordinate and two X-coordinates are given for a horizontal corridor or one X-coordinate and two Y-coordinates for a vertical corridor. The last line for each map contains the X and Y coordinates of the entry point (start and end point of travel) and the direction ('W' - left, 'E' - right, 'N' - up and 'S' - down). You may assume that: the entry point is not located at the cross-point of two corridors, and the explorer can always move forward in the direction given in the input. All coordinates are integers and do not exceed 32767 by absolute value and there are no more than 500 vertical corridors and no more than 500 horizontal corridors.

Output

For each map the program has to print the number of unvisited corridors (in a separate line).

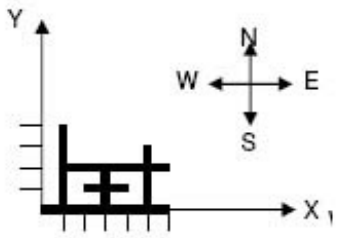
Example

Input:

```
2
6
H 0 6 0
H 2 1 6
V 1 0 4
V 5 3 0
V 3 0 2
H 1 2 4
6 0 W
1
V 0 -5 5
0 0 S
```

Output:

```
1
0
```



picture for first sample input