Robofence

Program Name: fence.java Input File: fence.in

You are a rich, lazy farmer. But just because you're a retired dot-com billionaire doesn't mean that you want to spend too much money hiring professionals to put up a fence on your property. So you've done what any reasonable farmer would do, built a 2-ton fencing robot from a slew of Lego Mindstorms sets.

The robot's programming is mostly complete, but you want to add one final feature.

Write a program that will review the robot's path and report the total number of yards of fencing used and the total area enclosed by the fence.

Input

The first line of input will consist of a single integer, *n*, indicating the number of datasets in the remaining input. Each dataset will consist of:

1. A line containing a single integer, 4 <= *m* <= 20, indicating how many straight stretches of fence the robot will be putting up.
2. A line containing *m* integers separated by *m*-1 letters. Each integer is the number of yards long that a stretch of fence will be. Each letter will indicate the direction the robot will turn between two stretches of fence (either 'L' – left, or 'R' – right). Turns are always exactly 90 degrees.

Fence laying instructions will always form an enclosed fence (i.e., the robot will always end up where it started), and the fence will never cross itself. Also, the entire fenced area will be coverable by a 50x50 square.

Output

For each dataset in the input, output the statement, "Fence X is Y yards long and encloses an area of Z square yard(s)." on its own line. X will be 1 for the first dataset, 2 for the second, etc. Y and Z are the values you will calculate.

Example Input File

```
2
4
2 L 2 L 2 L 2
8
3 L 6 L 3 L 2 R 2 L 2 L 2 R 2
```

Example Output To Screen

```
Fence 1 is 8 yards long and encloses an area of 4 square yard(s).
Fence 2 is 22 yards long and encloses an area of 22 square yard(s).
```