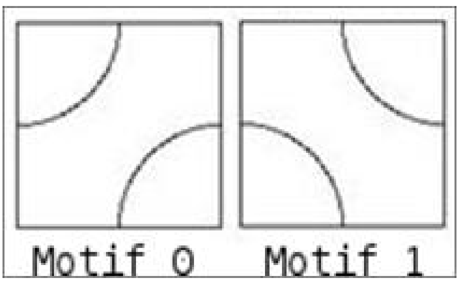
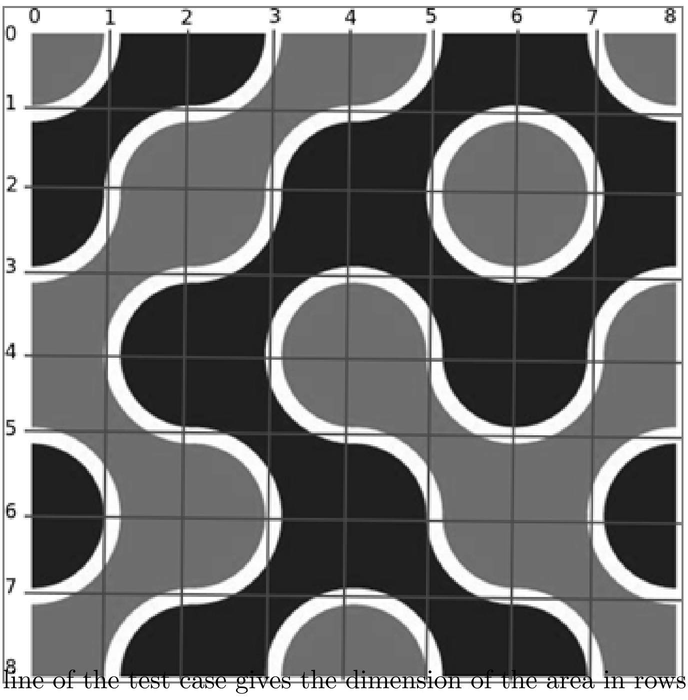


In 1704 French mathematician Sebastien Truchet proposed a tiling system that started with simple motifs and used it's rotations and random placement to create quite interesting tiling patterns. In this problem we will calculate the area enclosed by the curves in a special case of his tiling system.

Observe the two motifs here. Motif 0 is a square of length 2 units on each side. There are two circles with radius 1 unit drawn at two opposite corners of the square. Motif 1 is just a 90 degrees rotation of the first one (or you can think of it as drawing the circles on the other two corners).

Using just two basic motifs as shown here, we can tile an area to create rather interesting artistic patterns. They do not have to be symmetrically placed — we can lay out these motifs randomly to cover an area. If we then use a tool like paint bucket (found in most paint programs) at a particular point in the pattern, we can color a contiguous region with one color.

In this problem you'll be given a description of such a tiling pattern and then be asked how much area would it color if we use paint bucket starting at specific points.



Here is one illustration:

We have the motifs randomly placed on a 9×9 grid. Then we used a black color at position (2, 4) (Here 2 denotes the distance from the top and 4 denotes the distance from left). It colored all the contiguous region it found bounded by the grid's boundary and the curves themselves. We could achieve the same effect if we used the color at position (4, 6), (0, 6), (1, 7) etc. However, using the paint bucket on the perimeter of the curves (such as (3, 4)) will only color the perimeter line of the curves. They will not fill up a region with any color.

Input

The first line of the input starts with the number of test cases T ($1 \leq T \leq 100$). In the next T lines you get the description of T test cases. For each test case the first line of the test case gives the dimension of the area in rows and columns format. You can assume that rows R and columns C are at most 100 units long ($0 < R, C \leq 100$). For each of the next R lines you'll have the row description given in binary form. The basic motifs are numbered '0' and '1'. For example, the accompanying picture's first row can be described as '0001'. After the R row descriptions you'll be given Q ($1 \leq Q \leq 100$) — the number of queries. Each query will be of the form x ($0 \leq x \leq 2R$) and y ($0 \leq y \leq 2C$) where x and y are integers denoting the row and column number where you will use the paint bucket tool.

Output

For each test case print the test case number as '`Case` C`:`' in one line where C is the test case number. That line will be followed by Q lines of output for the queries on that test case. For each query print one number giving the area enclosed by the boundary and the curves that contain that point. If the point in question itself lies on a curve, you can assume the enclosed area to be zero. Your answer for each query must be rounded to 4 (four) digits after the decimal point. If the output is 0 (zero) make sure that you don't print it as '`-0.0000`'. Other than that inputs will be such that small precision error will not cause difference in output.

Sample Input

```
3
1 2
01
4
0 0
2 0
0 1
0 2
2 2
01
00
1
2 2
3 1
1
0
1
2
3 1
4 2
```

Sample Output

```
Case 1:
0.7854
4.8584
0.0000
4.8584
Case 2:
4.7854
Case 3:
7.2876
1.5708
```