

In this particular problem, *The Unreal Tournament* is a tournament, which consists of only two teams. Let these two teams be *Abahoni* and *Mohamedan*. They play in between them not more than $2n - 1$ games, the winner being the first team to achieve n victories. You can assume that there are no tied games, the result of each game is independent and for any match there is a constant probability p that team *Abahoni* will win and hence there is a constant probability $q = 1 - p$ that team *Mohamedan* will win.

$P(i, j)$ is the probability that team *Abahoni* will win the series given that they still need i more victories to achieve this, whereas team *Mohamedan* still need j more victories if they are to win. The $P(i, j)$ can be computed with a function like the following

```
Function P(i, j)
    if i = 0 then return 1
    else if j = 0 then return 0
    else return pP(i-1, j) + qP(i, j-1)
```

You will have to write a program that gives the probability of winning for any p , i and j and also gives the number of recursive calls required if the function above is used to get the probability $P(i, j)$.

Input

The input file contains several sets of input. The first line of a set contains one floating-point number $p(0 < p < 1)$, and an integer $N(0 \leq N < 1001)$ where p is the winning probability of *Abahoni* and N is the number queries to follow. Each of the next N lines contains two integers $i(0 \leq i \leq 1000)$ and $j(0 \leq j \leq 1000)$. Input is terminated by a set, which has zero as the value of N . This set should not be processed.

Output

For each query you should print two lines. The first line contains the value of $P(i, j)$ with five digits after the decimal and the second line contains a round number which is the number of recursive calls needed if the function mentioned above was used to determine the value of $P(i, j)$. If the value of $P(i, j)$ is undefined you should print '-1' as its value with similar formatting. A blank line should be printed between the outputs of two consecutive sets.

Sample Input

```
0.5 3
1 1
2 2
3 3
0.5 2
10 3
10 2
0.7 0
```

Sample Output

```
0.50000
2
0.50000
10
0.50000
38

0.01929
570
0.00586
130
```