Dexter Rank

Dexter just participated in a coding contest and is now waiting for judge to give final result . He is very bored of waiting every time for result and now wants to find his expected rank based on current scoreboard and chances of failure for problems. Formally there are M problems in the contest and there are N coders in the contest. Dexter knows that coder i has submitted problem j with penalty of penalty[i][j] . Dexter knows that problem j will pass with probability prob[j] (this also applies for Dexter himself also :)) . This is independent for each coder and problem. Now Dexter wants to know what would be his expected rank after system test.

Note: Coders are ranked based on number of correct solution, then total penalty for correct solutions. If there is tie even after that, all coders with same problems and penalty are given same rank.

Example : After system test if scoreboard is like this :

```
Coder 1 : 300  -1    -1          Score : 1  Penalty : 300
Coder 2 : 100  -1    200         Score : 2  Penalty : 300
Coder 3 : -1    300  -1          Score : 1  Penalty : 300
Coder 4 : -1    -1    400        Score : 1  Penalty : 400
```

Ranks would be : 2 1 2 4

INPUT :
The first line of input contains a single line **T** , which represents the number of test cases. F
For each test case first line contains two space separated integers **N** and **M** .
Then Next **N** lines contains **M** integers , j'th integer on i'th is penalty[i][j] which means penalty for i^{th} coder on problem j , if he submitted j'th problem, otherwise it is -1 .
Dexter is first coder in the list .
Then next line contains **M** space separated integers, where i^{th} integer denotes probability of passing problem i in system test, if submitted.

OUTPUT:
Print the expected rank of the Dexter after system test with exactly **4** decimal places.

CONSTRAINTS
1 <= **T** <= 16
2 <= **N** <= 256
1 <= **M** <= 12
1 <= penalty[i][j] <= 1000000 or penalty[i][j] = -1
0 <= prob[i] <= 100

EXAMPLE

INPUT:

3
2 1

100
150
50
2 2
-1 -1
10 -1
0 100
3 2
100 200
-1 199
99 -1
50 50

OUTPUT:
1.2500
1.0000
1.6250

EXPLANATION :
For sample case 1 , Dexter would have first rank if his solution passes or coder-2's solution fails system test which has probability 0.75 , otherwise he would have rank 2
answer = 1*0.75 + 0.25*2 = 1.2500

For sample case 2 , All coders would be on rank 1 with 0 problem solved and 0 penalty in all cases , so answer = 1.0000