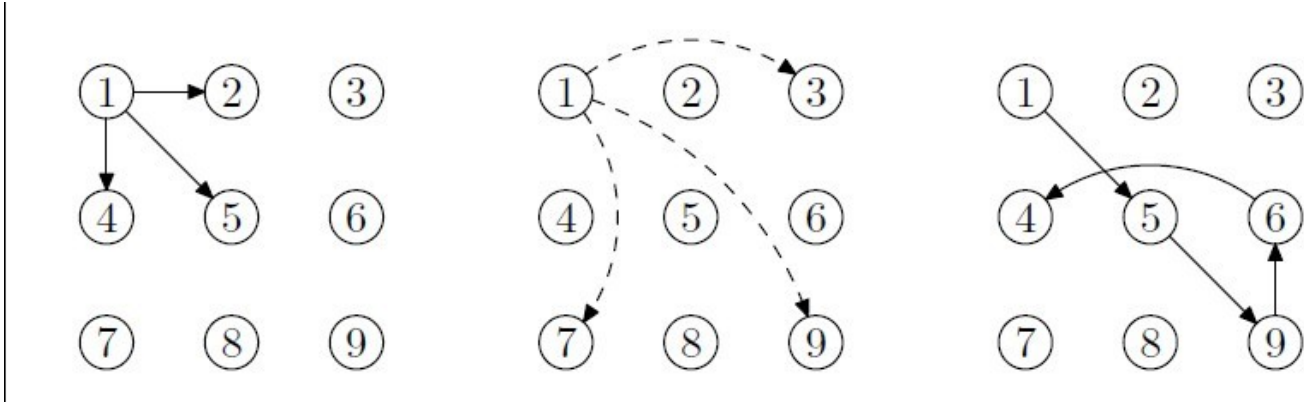


Unlock the Cellphone

Modern high-tech cell phones use unlock patterns to unlock the system. The pattern is usually a 3*3 dot array. By moving your finger over these dots, you can generate your personal unlock pattern. More specifically, press your finger over any starting dot, then slide all the way to the next dot, touch it, and so on. Jumping is not allowed. For example, starting from dot 1, you can slide to touch dot 2, dot 4 and dot 5, but sliding directly to dot 3, dot 7 or dot 9 are not allowed. Note that sliding from 1 to 6 and 8 is also allowed because they are not considered as jumping over any dot. However, you can jump a dot if it has been touched before. For example, starting with 1-5-9-6, you can slide directly to dot 4.



Here is a very particular cell phone. It has a dot array of size $n \times m$. Some of the dots are ordinary ones: you can touch, and slide over them when touched before; some are forbidden ones: you cannot touch or slide over them; some are inactive ones: you cannot touch them, but can slide over them. Each dot can only be touched once. You are required to calculate how many different unlock patterns passing through all the ordinary dots.

Input

The input contains several test cases. Each test case begins with a line containing two integers n and m ($1 \leq n, m \leq 5$), indicating the row and column number of the lock keypad. The following n lines each contain m integers $K_{i,j}$ indicating the properties of each key, $K_{i,j}=0$ stands for an ordinary key, $K_{i,j}=1$ stands for a forbidden key; and $K_{i,j}=2$ stands for an inactive key. The number of ordinary keys is greater than zero and no more than 16.

Input terminates by EOF.

Output

For each test, output an integer indicating the number of different lock patterns.

Example

Input:

```
2 2
0 0
0 0
3 3
0 0 0
0 2 1
0 0 0
```

Output:

```
24
2140
```