

## F. Treasure Map

Time limit: 1 second  
Memory limit: 65535 kBytes

### Description

#### The Treasure Map

In the quiet village of Numeria, an old cartographer is famous for his intricate maps and mysterious puzzles. Before retiring, he created his final treasure map, which, according to rumors, leads to a hidden trove of ancient coins. The map is unique. Instead of typical landmarks, it contains a grid of numbers. Each position in the grid holds a value that serves as a clue to locating the treasure. The villagers soon realized that to decode the map, they would need to apply a specific operation XOR on the coordinates. To determine the exact location of the treasure, they must find the  $K$ -th largest XOR value among all the grid's coordinates. This value is essential as it directly points to the location of the hidden treasure.

#### The Challenge

Help the villagers locate the treasure by writing a program to determine the  $K$ -th largest XOR coordinate value. Given an  $n \times m$  matrix of natural numbers and a number  $K$ , the program must compute the XOR for each coordinate and return the  $K$ -th largest value (1-indexed, meaning there is a first largest, not a "0-th" largest). The value at the matrix coordinate  $(a, b)$  is the XOR of all matrix elements  $(i, j)$  where  $0 \leq i \leq a < m$  and  $0 \leq j \leq b < n$  (indexes starting from 0).

The XOR (exclusive OR) operation is a logical operation that compares corresponding bits of two numbers and produces a result according to the following rule:

- The result bit is 1 if the bits being compared are different.
- The result bit is 0 if the bits being compared are the same.

### Input

The first line contains the values  $n$ ,  $m$  and  $K$ . The remaining  $n$  lines contain the rows of the matrix, with  $m$  elements on each line.

### Output

The output should contain a single number, the  $K$ -th largest XOR coordinate value.

### Constraints

- $1 \leq n, m \leq 1000$
- $0 \leq matrix[i][j] \leq 10^6$
- $1 \leq K \leq m * n$

### Example

Input	Output
2 2 1	7
5 2	
1 6	

*Explanation:* The result is 7, because that is the largest XOR value (the value at coordinate (0,1) is calculated as 5 XOR 2 = 7).

Input	Output
2 2 4	0
5 2	
1 6	

*Explanation:* The value at coordinate (1,1) is calculated as 5 XOR 2 XOR 1 XOR 6 = 0, which is the 4th largest value.