Quick Sum

Time limit: 1 sec

Given a 2D array **A** of **N** row and **M** columns of integer, we need to calculate the summation of every element within the range A[r1][c1] to A[r2][c2]. Formally, we define a *block summation of (r1,c1) and (r2,c2)* as $\sum_{i=r1}^{r2} \sum_{j=c1}^{c2} A[i][j]$, assuming 0 ≤ r1 ≤ r2 < N and 0 ≤ c1 ≤ c2 < M. Given **K** pairs of (r1,c1) and (r2, c2), calculate the block summation of each pair.

Obviously, solving this problem can be done easily in O(NM) for each block summation. However, we can speed this up by the following algorithm.

1. we define "prefix sum", **P(a,b)** as $\sum_{i=0}^{a} \sum_{j=0}^{b} A[i][j]$ when both **a** and **b** is non-negative and define **P(a,b)** as 0 when either **a** or **b** is negative.
2. The block summation of (r1,c1) and (r2,c2) can be computed directly from the prefix sum by the following formula, P(r2,c2) – P(r1-1,c2) – P(r2,c1-1) + P(r1-1,c1-1). Therefore, if we know the value of P(a,b), we can calculate the block summation in O(1).
3. We can pre-calculate P(a,b) for non-negative value of a and b can be done by the following algorithm. This has to be done only once.

```
Create a 2D array p which is used to store the value of P(a,b)
// p will be applicable for only non-negative value of a and b
p[0][0] = A[0][0]
for each column j of A from 1 to M-1
     p[0][j] = p[0][j-1] + A[0][j]
for each row i of A from 1 to N-1
     p[i][0] = p[i-1][0] + A[i][0]
     for each column j of A from 1 to M-1
        p[i][j] = p[i-1][j] + p[i][j-1] – p[i-1][j-1] + A[i][j]
```

Use the above algorithm to solve the problem of calculation of block summation.

Input

- The first line of input contains three integers **N, M** and **K** where 1 ≤ N,M ≤ 1000 and 1 ≤ K ≤ 100,000.
- The next N lines describe the array A. Each line representing the value in the array A for each row from row 0 to row N-1.
  - Each line contains M integer, representing the value in each row from column 0 to column M-1.
  - The value of each element in a is in the range 0 to 999.

- The next K lines describe K pair of (r1,c1) and (r2,c2) that we need to calculate the block summation. Each line contains 4 integers describing r1, c1, r2 and c2, respectively. It is guaranteed that $0 \le r1 \le r2 < N$ and $0 \le c1 \le c2 < M$.

Output

There must be exactly **K** lines. Each line must show the block summation of the given pairs of (r1, c1) and (r2, c2), starting from the first pair to the last pair respectively.

Example

Input	Output
3 5 7 1 2 3 4 5 6 6 6 6 6 7 7 3 1 1 0 0 0 0 0 0 1 1 1 1 2 1 1 1 2 3 1 0 2 2 0 1 2 2 0 0 2 4	1 15 13 29 35 27 64