

Count zeros in a sorted matrix

Given a **N X N** binary Square Matrix where each row and column of the matrix is sorted in ascending order. Find the total number of **zeros** present in the matrix.

Example 1:

Input: N = 3

```
A = {{0, 0, 0},
      {0, 0, 1},
      {0, 1, 1}}
```

 Output: 6

Explanation: The first, second and third row contains 3, 2 and 1 zeroes respectively.

Example 2:

Input: N = 2

```
A = {{1, 1},
      {1, 1}}
```

Output: 0

Explanation: There are no zeroes in any of the rows.

Your Task:

You don't need to read input or print anything. Your task is to complete the function **countZeros()** which takes a **2D matrix** as input and returns the number occurring only once.

Expected Time Complexity: $O(N)$.

Expected Auxiliary Space: $O(1)$.

Constraints

$0 < N \leq 10^3$

$0 \leq A[i] \leq 1$

```
def countZeroes(self, A, N):

    i = 0
    j = len(A[0])-1
    count = 0
    while i<len(A) and j>=0:
        temp = A[i][j]
        if temp ==0:
            count += j+1
            i = i+1
```

```
        else:  
            j = j-1  
    return count
```