**SOLUTION**

class Solution {

public:

Solution(){

ios::sync\_with\_stdio(false);

std::cin.tie(nullptr);

std::cout.tie(nullptr);

}

int countSquares(vector<vector<int>>& matrix) {

int m = matrix.size();

if(m == 0)

return 0;

int n = matrix[0].size();

if(n == 0)

return 0;

// vector<vector<int>> dp(m,vector<int>(n,0));

int total = 0;

for(int i=0;i<m;i++){

for(int j=0;j<n;j++){

if(matrix[i][j]){

if(i && j){

matrix[i][j] = min(matrix[i-1][j], min(matrix[i-1][j-1], matrix[i][j-1])) + 1;

}

total+=matrix[i][j];

}

}

}

return total;

}

};

**TIME COMPLEXITY= O(N\*M)**

**SPACE COMPLEXITY= O(1)**