Problem 1277: Count Square Submatrices with All Ones

Problem Information

Difficulty: Medium
Acceptance Rate: 80.64%
Paid Only: No
Tags: Array, Dynamic Programming, Matrix

Problem Description

Given a `m * n` matrix of ones and zeros, return how many **square** submatrices have all ones.

Example 1:

Input: matrix = [[0,1,1,1], [1,1,1,1], [0,1,1,1]] **Output:** 15 **Explanation:** There are **10** squares of side 1. There are **4** squares of side 2. There is **1** square of side 3. Total number of squares = 10 + 4 + 1 = **15**.

Example 2:

Input: matrix = [[1,0,1], [1,1,0], [1,1,0]] **Output:** 7 **Explanation:** There are **6** squares of side 1. There is **1** square of side 2. Total number of squares = 6 + 1 = **7**.

Constraints:

* `1 <= arr.length <= 300` * `1 <= arr[0].length <= 300` * `0 <= arr[i][j] <= 1`

Code Snippets

C++:

```
class Solution {
public:
```

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

}
};
```

Java:

```
class Solution {
public int countSquares(int[][] matrix) {

}
}
```

Python3:

```
class Solution:
def countSquares(self, matrix: List[List[int]]) -> int:
```