

Maximal Square

Solution

Given a 2D binary matrix filled with 0's and 1's, find the largest square containing only 1's and return its area.

Example:

Input:

```
1 0 1 0 0
1 0 1 1 1
1 1 1 1 1
1 0 0 1 0
```

Output: 4

Java



```
1 class Solution {
2     public int maximalSquare(char[][] matrix) {
3
4         if(matrix.length == 0) return 0;
5
6         int max = 0;
7         int[][] dp = new int[matrix.length][matrix[0].length];
8         for(int i = 0; i < dp.length; i++) {
9             dp[i][0] = matrix[i][0] == '1' ? 1 : 0;
10            if(dp[i][0] == 1) max = 1;
11        }
12
13        for(int j = 0; j < dp[0].length; j++) {
14            dp[0][j] = matrix[0][j] == '1' ? 1 : 0;
15            if(dp[0][j] == 1) max = 1;
16        }
17
18        for(int i = 1; i < dp.length; i++) {
19            for(int j = 1; j < dp[0].length; j++) {
20                if(matrix[i][j] == '0') dp[i][j] = 0;
21                else {
22                    dp[i][j] = Math.min(dp[i-1][j-1], Math.min(dp[i][j-1], dp[i-1][j])) + 1;
23                    max = Math.max(max, dp[i][j]);
24                }
25            }
26        }
27
28        return max * max;
29    }
30 }
31 }
```

☐ Custom Testcase ([Contribute](#))


Run Code

Submit