Problem 1820: Maximum Number of Accepted Invitations

Problem Information

Difficulty: Medium
Acceptance Rate: 51.95%
Paid Only: Yes
Tags: Array, Depth-First Search, Graph, Matrix

Problem Description

There are `m` boys and `n` girls in a class attending an upcoming party.

You are given an `m x n` integer matrix `grid`, where `grid[i][j]` equals `0` or `1`. If `grid[i][j] == 1`, then that means the `ith` boy can invite the `jth` girl to the party. A boy can invite at most**one girl** , and a girl can accept at most **one invitation** from a boy.

Return _the**maximum** possible number of accepted invitations._

Example 1:

Input: grid = [[1,1,1], [1,0,1], [0,0,1]] **Output:** 3**Explanation:** The invitations are sent as follows: - The 1st boy invites the 2nd girl. - The 2nd boy invites the 1st girl. - The 3rd boy invites the 3rd girl.

Example 2:

Input: grid = [[1,0,1,0], [1,0,0,0], [0,0,1,0], [1,1,1,0]] **Output:** 3 **Explanation:** The invitations are sent as follows: -The 1st boy invites the 3rd girl. -The 2nd boy invites the 1st girl. -The 3rd boy invites no one. -The 4th boy invites the 2nd girl.

Constraints:

* `grid.length == m` * `grid[i].length == n` * `1 <= m, n <= 200` * `grid[i][j]` is either `0` or `1`.

Code Snippets

C++:

```
class Solution {
public:
int maximumInvitations(vector<vector<int>>& grid) {

}
};
```

Java:

```
class Solution {
public int maximumInvitations(int[][] grid) {

}
}
```

Python3:

```
class Solution:
def maximumInvitations(self, grid: List[List[int]]) -> int:
```