Maximum Stone Removal

There are **n** stones at some integer coordinate points on a 2D plane. Each coordinate point may have at most one stone.

You need to remove some stones.

A stone can be removed if it shares either the same row or the same column as another stone that has not been removed.

Given an array stones of length **n** where **stones[i] = [xi, yi]** represents the location of the **ith** stone, return the **maximum** possible number of stones that you can remove.

Example 1:

Input:
n=6
[[0 0] ,[0 1], [1 0] ,[1 2] ,[2 1] ,[2 2]]

Output:
5

Example:
One way to remove 5 stones are
1--[0,0]
2--[1,0]
3--[0,1]
4--[2,1]
5--[1,2]

Your Task:

You don't need to print or input anything. Complete the function **maxRemove()** which takes an integer array arr, an integer n as the input parameters and returns an integer, denoting the maximum number of stones that can be removed.

Expected Time Complexity: O(N+K)

Expected Space Complexity: O(K)

Constraints:

$1 <= n <= 1000$
$0 <= x[i], y[i] <= 10^4$
No two stones are at same position.