3238. Find the Number of Winning Players

You are given an integer n representing the number of players in a game and a 2D array pick where pick[i] = [xi, yi] represents that the player xi picked a ball of color yi.

Player i **wins** the game if they pick **strictly more** than i balls of the **same** color. In other words,

* Player 0 wins if they pick any ball.
* Player 1 wins if they pick at least two balls of the *same* color.
* ...
* Player i wins if they pick at leasti + 1 balls of the *same* color.

Return the number of players who **win** the game.

**Note** that *multiple* players can win the game.

**Example 1:**

**Input:** n = 4, pick = [[0,0],[1,0],[1,0],[2,1],[2,1],[2,0]]

**Output:** 2

**Explanation:**

Player 0 and player 1 win the game, while players 2 and 3 do not win.

**Example 2:**

**Input:** n = 5, pick = [[1,1],[1,2],[1,3],[1,4]]

**Output:** 0

**Explanation:**

No player wins the game.

**Example 3:**

**Input:** n = 5, pick = [[1,1],[2,4],[2,4],[2,4]]

**Output:** 1

**Explanation:**

Player 2 wins the game by picking 3 balls with color 4.

**Constraints:**

* 2 <= n <= 10
* 1 <= pick.length <= 100
* pick[i].length == 2
* 0 <= xi <= n - 1
* 0 <= yi <= 10

class Solution {

    public int winningPlayerCount(int n, int[][] pick) {

          int[][] playerColorMatrix = new int[n][11];

        // initialize each possible player

        for (int i = 0; i < n; i++) {

            int[] zeroInitializer = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};

            playerColorMatrix[i] = zeroInitializer;

        }

        //System.out.println("INITIAL PLAYER COLOR MATRIX");

        //printPlayerColorMatrix(playerColorMatrix, n);

        // Process pick

        int rows = pick.length;

        for (int r=0; r < rows; r++) {

            int playerNumber = pick[r][0];

            int playerColor = pick[r][1];

           // System.out.println("Player: " + playerNumber + " Color: " + playerColor);

            playerColorMatrix[playerNumber][playerColor] += 1;

            //printPlayerColorMatrix(playerColorMatrix, n);

        }

        //System.out.println("FINAL PLAYER COLOR MATRIX");

        //printPlayerColorMatrix(playerColorMatrix, n);

        // Count the number of winning players

        int winningPlayers = 0;

        for (int i = 0; i < n; i++) {

            for (int j = 0; j < 11; j++) {

                if (playerColorMatrix[i][j] > i) {

                    winningPlayers += 1;

                    break;

                }

            }

        }

        return winningPlayers;

    }

}