class Solution {

public:

    vector<vector<int>> imageSmoother(vector<vector<int>>& img) {

        int m = img.size();

        int n = img[0].size();

        vector<vector<int>>res=img;

        for (int i = 0; i < m; i++) {

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

                int up = 0;

                int left = 0;

                int right = 0;

                int down = 0;

                int upleft = 0;

                int upright = 0;

                int downleft = 0;

                int downright = 0;

                int cnt = 1;

                int curr=img[i][j];

                if (i - 1 >= 0) {

                    up = img[i - 1][j];

                    cnt++;

                }

                if (j - 1 >= 0) {

                    left = img[i][j - 1];

                    cnt++;

                }

                if (j + 1 < n) {

                    right = img[i][j + 1];

                    cnt++;

                }

                if (i + 1 < m) {

                    down = img[i + 1][j];

                    cnt++;

                }

                if (i - 1 >= 0 && j - 1 >= 0) {

                    upleft = img[i - 1][j - 1];

                    cnt++;

                }

                if (i - 1 >= 0 && j + 1 < n) {

                    upright = img[i - 1][j + 1];

                    cnt++;

                }

                if (i + 1 < m && j - 1 >= 0) {

                    downleft = img[i + 1][j - 1];

                    cnt++;

                }

                if (i + 1 < m && j + 1 < n) {

                    downright = img[i + 1][j + 1];

                    cnt++;

                }

                res[i][j] = (curr+up + left + right + down + upleft + upright +

                             downleft + downright) /

                            cnt;

            }

        }

        return res;

    }

};

s