// Nombre del alumno .....Jorge del Valle Vazquez

// Usuario del Juez ......DG206

#include <iostream>

#include <iomanip>

#include <fstream>

#include <vector>

using tmatriz = std::vector <std::vector<int>>;

// funcio?n que resuelve el problema

std::vector<int> resolver(tmatriz const & a) {

std::vector<int>sol;

for (int f = 0; f < a.size(); ++f) {

int min = a[f][0];

for (int c = 0; c < a[f].size(); ++c) {

if (a[f][c] < min) {

min = a[f][c];

}

}

sol.push\_back(min);

}

return sol;

}

void leer(tmatriz & matriz) {

for (int f = 0; f < matriz.size(); ++f)

for (int c = 0; c < matriz[f].size(); ++c)

std::cin >> matriz[f][c];

}

std::ostream & operator<<(std::ostream& salida ,std::vector<int>const& v) {

for (int i = 0; i < v.size(); ++i) {

std::cout << v[i] << ' ';

}

std::cout << '\n';

return salida;

}

// Resuelve un caso de prueba, leyendo de la entrada la

// configuracio?n, y escribiendo la respuesta

bool resuelveCaso() {

// leer los datos de la entrada

int f1 = 0; int c1 = 0;

std::cin >> f1;

if (!std::cin)

return false;

std::cin >>>> c1;

tmatriz m1(f1, std::vector<int>(c1));

leer(m1);

if (!std::cin)

return false;

std::vector<int> sol = resolver(m1);

// escribir sol

std::cout << sol;

return true;

}

int main() {

// Para la entrada por fichero.

// Comentar para acepta el reto

#ifndef DOMJUDGE

//std::ifstream in("datos1.txt");

//auto cinbuf = std::cin.rdbuf(in.rdbuf()); //save old buf and redirect std::cin to casos.txt

#endif

while (resuelveCaso())

;

// Para restablecer entrada. Comentar para acepta el reto

#ifndef DOMJUDGE // para dejar todo como estaba al principio

//std::cin.rdbuf(cinbuf);

system("PAUSE");

#endif

return 0;

}