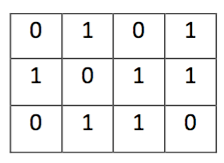
Buscaminas

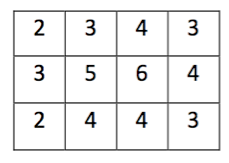
|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |

**Descripción**

Todos hemos jugado el clásico juego de computadora conocido como "Buscaminas". Pues bien, este problema está inspirado en dicho juego, pero, por supuesto, con pequeñas variaciones. Para empezar, comenzaremos con una matriz cuyos valores solo pueden ser 0 o 1, como esta:



A partir de esta matriz, debes generar una segunda matriz (de las mismas dimensiones) a la cual denotaremos por B, en la que el número contenido en cada casilla representará el número de bombas con las cuales está en contacto esa casilla en específico (incluyéndose a sí misma).



**Problema**

Considerar una matriz a la que denotaremos por A de M×N (1≤M,N≤200) con M filas y Ncolumnas cuyos valores solo pueden ser 0 o 1, a partir de esta se genera una segunda matriz a la que denotaremos por B de las mismas dimensiones, en la que cada casilla X de B tiene una valor entre 0 y9 inclusive, que es igual a la suma de las casillas de la matriz A que son adyacentes a la casilla X en Amás el valor de la casilla X en A (dos casillas son adyacentes si tienen un vértice en común). Genera la matriz B a partir de la matriz A.

**Entrada**

En la primera linea dos numeros M y N separados por un unico espacio que nos indica el tamaño de la matriz. En las siguientes M lineas habrá por linea N numeros separados por un espacio entre ellos que seran los datos de la matriz

**Salida**

M líneas cada una con N numeros separados por un espacio entre ellos que nos demuestra la cantidad de bombas que tiene en contacto cada una de las casillas.

**Ejemplo**

| **Entrada** | **Salida** |
| --- | --- |
| 3 4  0 1 0 1  1 0 1 1  0 1 1 0 | 2 3 4 3  3 5 6 4  2 4 4 3 |

*Fuente: CONACUP*

Problema subido por: [conacup](https://omegaup.com/profile/conacup/)

| **ID** | **Status** | **Porcentaje** | **Penalty** | **Lenguaje** | **Memoria** | **Tiempo** | **Detalles** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| [Nuevo envío](https://omegaup.com/arena/problem/buscaminas1#problems/new-run) | | | | | | | | |
| 7719fdd8 | Respuesta correcta | 100.00% | 0 | cpp | 3.07 | 0.03 |  |  |
| **Envíos** | | | | | | | |

<https://omegaup.com/arena/problem/buscaminas1#problems>

#include <iostream>

#include <stdio.h>

#include <vector>

#include <conio.h>

using namespace std;

int main() {

    int M, N;

    scanf("%d %d", &M, &N);

    int A[M][N];

    for(int i =0; i < M; i++) {

        for(int j =0 ; j <N; j++) {

            int elem;

            scanf("%d", &elem);

            A[i][j] = elem;

        }

    }

    int B[M][N];

    for (int i = 0; i < M; i++)

    {

        for (int j = 0; j < N; j++)

        {

            int cont = 0;

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

            {

                if (A[i - 1][ j - 1] == 1)

                {

                    cont++;

                }

            }

            if (i - 1 >= 0)

            {

                if (A[i - 1][j] == 1)

                {

                    cont++;

                }

            }

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

            {

                if (A[i - 1][j + 1] == 1)

                {

                    cont++;

                }

            }

            if (j - 1 >= 0)

            {

                if (A[i][j - 1] == 1)

                {

                    cont++;

                }

            }

            if (A[i][j] == 1)

            {

                cont++;

            }

            if (j + 1 < N)

            {

                if (A[i][j + 1] == 1)

                {

                    cont++;

                }

            }

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

            {

                if (A[i + 1][j - 1] == 1)

                {

                    cont++;

                }

            }

            if (i + 1 < M)

            {

                if (A[i + 1][j] == 1)

                {

                    cont++;

                }

            }

            if (i + 1 < M && j + 1 < N)

            {

                if (A[i + 1][j + 1] == 1)

                {

                    cont++;

                }

            }

            B[i][ j] = cont;

        }

    }

    for (int i = 0; i < M; i++)

    {

        for (int j = 0; j < N; j++)

        {

            //Console.Write(B[i, j] + " ");

            printf("%d ", B[i][j]);

        }

        printf("**\n**");

    }

    getch();

    return 0;

}