

---

**Matrices dispersas (1): conversión****Y29996\_es**

---

Una **matriz dispersa** (o esparcida) es una matriz en la que la mayoría de los elementos son cero. En lugar de almacenar todos los elementos, la representación dispersa guarda solo los elementos no nulos junto con sus posiciones, ahorrando memoria y tiempo de cómputo. Por ejemplo, la matriz:

$$\begin{pmatrix} 0 & 3 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 1 & 0 & 0 & 7 \end{pmatrix}$$

en representación dispersa por filas queda:

Fila 0: (1, 3) .  
Fila 1: (2, 5) .  
Fila 2: (0, 1) (3, 7) .

donde cada par (col, val) indica la columna y el valor de cada elemento no nulo, y el punto final . marca el final de la fila (incluso cuando la fila está vacía).

En este problema, representamos una matriz de enteros con el tipo `Matriu` y la versión dispersa con el tipo `MatriuEsparsa`, definidos en `matrius.hh`:

```
typedef vector<vector<int>> Matriu;

struct Casella {
    int pos;    // índice de columna
    int valor;  // valor del elemento
};

struct MatriuEsparsa {
    int ncols;           // número de columnas
    vector<list<Casella>> files; // filas de la matriz dispersa
};
```

Escribe la siguiente función:

```
/**
 * @brief Convierte una Matriu en una MatriuEsparsa.
 *
 * @param M Matriz rectangular de tamaño n x m.
 *
 * @pre M puede ser cualquier matriz rectangular (incluyendo vacía).
 * @post Devuelve una nueva MatriuEsparsa S, con S.ncols == m y S.files.size() =
 *       donde S.files[i] contiene, ordenadas por columna ascendentemente, las c
 *       {pos, valor} de los elementos no nulos de la fila i de M.
 */
MatriuEsparsa matriu_converteix_a_esparsa(const Matriu& M);
```

## Observación

Los ficheros públicos (icono del gatito) contienen:

<code>main.cc</code>	el programa principal, con la entrada/salida hecha
<code>matrius.hh</code>	los tipos <code>Matriu</code> , <code>Casella</code> i <code>MatriuEsparsa</code>
<code>Makefile</code>	para compilar con <code>make</code> cómodamente
<code>.vscode</code>	carpeta para compilar y depurar con VSCode

Hay que enviar únicamente la implementación de la función `matriu_converteix_a_esparsa`, incluyendo el include de `matrius.hh`. No hace falta enviar el `main`.

## Entrada

La primera línea contiene dos enteros  $n$  y  $m$  ( $0 \leq n, m \leq 1000$ ), el número de filas y columnas de la matriz. A continuación hay  $n$  líneas, cada una con  $m$  enteros separados por espacios que representan las filas de la matriz.

## Salida

La primera línea contiene dos enteros  $n$  y  $m$ , las dimensiones de la matriz dispersa resultante. A continuación hay  $n$  líneas, cada una con las casillas no nulas de la fila, en orden ascendente de columna, con el formato `(col, val)` separadas por espacios, seguidas de un punto `..`. Las filas sin ningún elemento no nulo se muestran únicamente con el punto `..`.

### Ejemplo de entrada 1

```
3 4
0 3 0 0
0 0 5 0
1 0 0 7
```

### Ejemplo de salida 1

```
3 4
(1, 3) .
(2, 5) .
(0, 1) (3, 7) .
```

### Ejemplo de entrada 2

```
4 4
1 0 0 2
0 0 0 0
0 -3 0 0
5 0 6 0
```

### Ejemplo de salida 2

```
4 4
(0, 1) (3, 2) .
.
(1, -3) .
(0, 5) (2, 6) .
```

## Información del problema

Autoría: Pau Fernández

Traducción: Pau Fernández

Generación: 2026-02-25T09:47:46.532Z

© Jutge.org, 2006–2026.

<https://jutge.org>