

Contexto

Estás trabajando de informático en una empresa que se dedica a reparar maquinaria pesada. Cada mecánico puede reparar ciertos tipos de avería. La reparación de una avería requiere de la asignación de un mecánico, y cada mecánico puede estar asignado cada día a un máximo de una avería. La empresa te pide que diseñes un programa que permita optimizar la asignación diaria de mecánicos a averías, de modo que se consiga reparar el máximo posible de averías cada día.

El Problema

Tenemos M mecánicos y A averías por reparar. Una tabla bidimensional C , de elementos booleanos $c(i,j)$, indica que el mecánico i tiene capacidad para reparar la avería j .

Se debe realizar una asignación de mecánicos a averías que maximice el número de averías reparadas en el día, teniendo en cuenta que cada mecánico puede ser asignado a una única avería.

Se admitirá cierto margen en la solución respecto del óptimo: cualquier solución que alcance el 60% del óptimo será considerada válida.

Entrada

La primera línea de la entrada contiene un entero, P , que indica el número de casos de prueba.

Cada caso de prueba contiene $1+M$ líneas:

- la primera línea tiene dos enteros, M y A , que indican el número de mecánicos y de averías, respectivamente;
- las siguientes M líneas contienen A enteros cada una, separados por espacios en blanco, correspondiendo el entero j -ésimo de la línea i a la capacidad del mecánico i para reparar la avería j , es decir, el elemento $c(i,j)$ de la tabla C . Un 1 indica que el mecánico i puede arreglar la avería j , y un 0 indica que no.

Salida

La primera línea de la salida contiene un entero, P , que indica el número de casos de prueba (es decir, el mismo que el de entrada).

Para cada caso, la salida deberá tener dos líneas:

- la primera línea es el número total de averías reparadas.
- la segunda línea indica qué mecánico se asignó a cada avería. Contendrá A números enteros, separados por espacios en blanco, donde el entero j -ésimo indica el mecánico asignado a la avería j ; para las averías no reparadas se utilizará el número 0.

Ejemplo de Entrada

```
3
2 3
0 1 0
0 0 0
4 3
1 0 1
1 1 0
0 0 0
1 1 1
2 3
0 0 1
1 0 1
```

Ejemplo de Salida

```
3
1
0 1 0
```

3
1 2 4
2
2 0 1

Ejemplos Extendidos

[Entrada](#) [Salida](#)
