

Satisfacción en el trabajo

En una empresa se desea distribuir n trabajos a n empleados. Todos los trabajos deben realizarse y ningún empleado debe estar ocioso. La empresa dispone de una tabla B de beneficios y otra S de satisfacción, donde $B[i, j]$ indica la cantidad que gana la empresa si el empleado i realiza la tarea j y $S[i, j]$ indica el nivel de satisfacción (> 0) que el empleado i consigue al ejecutar la tarea j . Los empleados de esta empresa son muy envidiosos y la empresa debe tener en cuenta a la hora de asignar los trabajos que un empleado se irá de la empresa si se da el caso de que en una planificación hay algún otro empleado con al menos el doble de satisfacción que él. Diseñar un algoritmo de **vuelta atrás** que encuentre la planificación con beneficio máximo para la empresa suponiendo que esta no quiere que ningún empleado se vaya.

Entrada

En la primera línea el número de casos de prueba. Para cada caso de prueba la primera línea es el número de tareas y empleados. A continuación aparecen la matriz de satisfacciones y la de beneficios por ese orden.

Salida

Por cada caso de prueba aparecerá una línea independiente con el beneficio de la mejor solución encontrada o bien el mensaje "Sin solucion factible" en el caso de que no haya ninguna.

Entrada de ejemplo

```
2
2
1 2
1 2

4 2
9 1

5
1 2 1 2 2
2 3 2 1 2
2 1 6 3 1
1 2 3 3 2
1 3 2 3 3

2 1 3 5 4
4 5 6 2 2
7 8 9 3 3
2 2 5 6 2
9 8 3 4 5
```

Salida de ejemplo

```
Sin solucion factible
31
```