

Fundamentos de Algoritmia

Grupos B, G, H e I

CONVOCATORIA EXTRAORDINARIA

CURSO 2024-2025

Ejercicio 3 (4 puntos)

Un tablero de ajedrez de tamaño $n \times n$ tiene en cada casilla una determinada cantidad de monedas. Queremos colocar n torres en el tablero, de forma que no se ataquen entre ellas y obtengamos la mayor cantidad posible de monedas, asumiendo que una colocación consigue las monedas que haya en las celdas en las que situamos las piezas.

Diseña un algoritmo de **vuelta atrás** que reciba el número de torres a colocar y la cantidad de monedas de cada casilla, y permita averiguar cuál es esa cantidad máxima de monedas.

Se valorará positivamente el uso de estrategias para podar el espacio de soluciones, sin afectar a la optimalidad del algoritmo.

Recuerda que en el ajedrez una *torre* solo puede moverse en horizontal o vertical.

Entrada

La entrada consiste en una serie de casos de prueba, cada uno de los cuales consta de:

- Una primera línea indicando el número de torres que hay que colocar, $n \leq 10$.
- Las monedas almacenadas en cada celda (entre 0 y 1000), distribuidas en n líneas de n números cada una.

Los casos de prueba finalizan con un 0.

Salida

Para cada caso de prueba se escribirá la máxima cantidad de monedas que se pueden conseguir.

Ejemplo

Entrada de ejemplo	Salida de ejemplo
3	8
1 3 0	13
4 0 1	
1 1 1	
2	
10 10	
3 2	
0	