

Cubo de luces*Contribución de Pablo Heiber***Descripción del problema**

Agustina tiene un cubo de $n \times n \times n$, formado por n^3 cubitos unitarios idénticos. Cada cubito unitario tiene una luz que puede estar encendida o apagada.

Se tiene además, por cada prisma de $1 \times 1 \times n$ del cubo (de los cuales hay $3n^2$, n^2 en cada una de las 3 direcciones posibles) un botón que cambia de estado esas n luces del prisma (las apagadas se encienden, y las encendidas se apagan).

Dada una configuración inicial, tu tarea consiste en indicarle a Agustina la mínima cantidad de pulsaciones de botones que deja todas las luces encendidas. Si no es posible lograrlo, debes indicarlo con -1 .

Detalles de implementación

Debes implementar la función:

- `cubo(v)`:
 - v : arreglo tridimensional (arreglo de arreglos de arreglos de enteros), de modo que $v[i][j][k]$ será 1 si la luz en la posición (i, j, k) del cubo está encendida, o 0 si está apagada.
 - Debe retornar un entero con la mínima cantidad de pulsaciones necesarias para encender todas las luces, o -1 si es imposible.

Cotas

- $3 \leq n \leq 8$

Evaluador local

El evaluador local lee de la entrada estándar:

1. Una línea con el entero n .
2. n bloques de n líneas cada uno. Cada línea contiene n enteros, separados por un espacio. El entero k de la línea j del bloque i indica el valor $v[i][j][k]$.

Escribe por la salida estándar el resultado de aplicar la función `cubo(v)`.

Ejemplo

Si el evaluador local recibe:

```
4
0 0 0 0
1 0 0 0
0 0 0 0
1 0 0 0
0 0 0 0
1 0 0 0
0 0 0 0
1 0 0 0
1 0 0 0
0 0 0 0
1 0 0 0
0 0 0 0
0 0 0 0
1 0 0 0
0 0 0 0
0 0 0 0
1 0 0 0
```

Con una solución correcta, imprime:

```
15
```

Si en cambio la entrada fuera:

3

0 0 0

0 1 0

0 0 0

0 0 0

0 0 0

0 0 0

0 0 0

0 0 0

0 0 0

Con una solución correcta, imprime:

Subareas

1. $v[i][j][k_1] = v[i][j][k_2]$ para todos los posibles i, j, k_1, k_2 (10 puntos)
2. $n = 3$ (21 puntos)
3. $n = 4$ (22 puntos)
4. $n = 5$ (30 puntos)
5. Sin más restricción (17 puntos)