

Back-track to the EDA

Equipo Sort

Al final no voy a llegar al máximo de gigavatios en cada tema de EDA. ¡No seré un McFly experto! Si lo hubiera sabido, habría hecho más misiones de los temas en los que me faltan y menos en los que me sobran. Claro, que también podría viajar al pasado y avisar a mi yo de septiembre de las misiones mínimas que tengo que hacer para conseguir el 10.

Además de las misiones estándar por cada tema, tenemos unas misiones extra que se pueden asignar a cualquier tema (pero a uno solo por misión) en el que vaya floja la puntuación: las *misiones a la carta*.

No puedo permitirme hacer más misiones de las necesarias, porque en 2º de Matemáticas e Informática tenemos 7 asignaturas y tengo que sacar tiempo de debajo de las piedras. Por otro lado, supongo que tardo siempre lo mismo en hacer cada misión: al final siempre acabo dedicando una tarde entera. Por eso me interesa hacer el mínimo número de misiones. También voy a suponer que voy a conseguir todos los gigavatios de las misiones que haga; ¡les voy a dedicar muchísimo tiempo!

La profesora es muy buena y siempre da misiones suficientes como para llegar a ese 10. Otra cosa es que nos cueste sudor y lágrimas (o una segunda matrícula en Cálculo Diferencial).

Entrada

La entrada estará compuesta por distintos casos de prueba. En la primera línea aparecen tres números: el número de gigavatios que necesito alcanzar en cada tema ($0 < G \leq 10^9$), el número de temas en el cuatrimestre ($0 < T \leq 10$) y el número de misiones por cada tema ($0 < M \leq 10$).

Por cada línea de las T siguientes líneas se encuentran M enteros, que corresponden a los gigavatios asignados a cada misión de dicho tema. Una misión puede no estar asignada en algún tema, en cuyo caso se me concedería cero gigavatios por hacerla. Además ninguna misión resta gigavatios, ¡encima de hacer el esfuerzo!

Tras esto, una última línea contiene M enteros, que corresponden a los gigavatios asignados a *misiones a la carta*.

La entrada termina con una línea con tres ceros que no debe procesarse.

Salida

Por cada caso de prueba se escribirá el número mínimo de misiones que tengo que realizar para llegar a G gigavatios en cada tema.

Entrada de ejemplo

```
1 1 1
1
0
4 2 2
1 1
3 1
2 0
20 2 7
0 9 2 2 5 3 0
4 8 2 2 0 4 0
0 0 0 0 0 0 0
20 2 7
0 9 2 2 5 3 0
4 8 2 2 0 4 0
5 0 0 0 0 0 0
20 6 7
0 9 2 2 5 3 0
4 8 2 2 0 4 0
4 8 2 0 0 6 3
4 14 2 0 0 4 3
4 8 2 0 10 6 3
4 8 2 0 0 6 3
0 0 0 0 0 0 0
0 0 0
```

Salida de ejemplo

```
1
5
10
9
24
```

Solución de búsqueda exhaustiva

En `manostijeras.cpp` se resuelve el problema guardando en `gigasxtema` los gigavatios acumulados por cada tema, y recorriendo toda la matriz:

- Si estamos en un nodo dentro de un tema (fila), se llama a la función SIN ese nodo y CON ese nodo (tras haberlo incluido en `gigasxtema` y haber incrementado `solparcial`), llamando a la misión (columna) siguiente.
- Si hemos acabado el tema, se llama a la función con el tema siguiente y el número de misión a 0.
- Si hemos acabado la matriz, quedan por añadir las posibles *misiones a la carta*. Entonces se llama a `final`.

En `final` volvemos a probar todas las opciones añadiendo o no añadiendo cada *misión a la carta* a cada uno de los temas mediante el bucle.

Si hemos acabado con todas las misiones, se comprueba si el estado actual es solución (si la solución parcial es menor que la total y si en todos los temas se ha llegado al máximo `g`).

Poda 1: Podar los ceros

En `manostijeras1.cpp` se podan todos los nodos no positivos, ya que solo añaden volumen de misiones sin aportar gigavatios. Se podan tanto las misiones estándar como las *a la carta*.

Poda 2: Podar los que se pasan de gigavatios

En `manostijeras2.cpp` se podan (además de los anteriores) todos los nodos que sobrepasen el número de gigavatios requerido `g`, ya que una vez se pasa de `g` en un tema, no hacen falta más misiones. Se podan tanto las misiones estándar como las *a la carta*.

Poda 3: Podar los peores que lo que ya tenemos

En `manostijeras3.cpp` se podan (además de los anteriores) todos los nodos en los que la solución parcial (número de misiones) sea superior a la solución total, ya que a partir de ese punto solo se van a añadir más misiones y ya tenemos una solución mejor. Se podan tanto las misiones estándar como las *a la carta*.

Poda 4: Podar los peores que la forma polinómica

En `manostijeras4.cpp` acotamos mejor la solución total: nos damos cuenta de que hay una manera de resolverlo de manera polinómica si no existieran las

misiones a la carta, por lo que, si hay solución polinómica, nuestra solución mínima será menor o igual a ésta. Por tanto, se inicializa `sol` a este resultado y, en su defecto, al límite superior de los enteros (lo que había antes).