

3. (4 puntos) La Comunidad de FALuca ha sufrido una catástrofe natural que ha afectado a  $m \geq 1$  poblaciones. Es necesario distribuir ayuda a dichas poblaciones, cada una de las cuales ha comunicado sus necesidades (kilos de ayuda) al jefe de protección civil de la comunidad. Se dispone de  $n \geq 1$  voluntarios para repartir las ayudas y de una tabla que indica para cada voluntario, cuántos kilos de ayuda puede llevar a cada una de las poblaciones (esto depende de muchos factores como la accesibilidad de la población, la localización y los medios de transporte del voluntario etc.) A una población puede enviarse cualquier número de voluntarios pero cada voluntario solo puede ir a una población. Como no se sabe cuando cada voluntario será capaz de llegar hasta la población que se le asigne, se les ha pedido que cada uno lleve los todos los kilos especificados en la tabla. Si luego la cantidad de ayuda que los voluntarios llevan a una población excede sus necesidades solo se repartirá la notificada como necesaria, y en caso contrario se repartirá toda la que han llevado los voluntarios. El objetivo es maximizar la cantidad de kilogramos de ayuda repartida en las poblaciones teniendo en cuenta que el jefe de protección civil quiere que al menos  $l$  poblaciones ( $l \geq 0$ ,  $l \leq n$  y  $l \leq m$ ) reciban la ayuda que solicitaron.
- Define el *espacio de soluciones* e indica cómo es el *árbol de exploración*.
  - Implementa un algoritmo de *vuelta atrás* que resuelva el problema. Explica claramente los *marcadores* que has utilizado.
  - Plantea dos posibles funciones de poda de optimalidad, razona sobre cual de ellas es mejor e impleméntala en tu algoritmo.

## Entrada

La entrada comienza con una línea que contiene el número de casos de prueba. Cada caso de prueba contendrá el valor del número de personas  $n$ , de poblaciones  $m$  y el número de poblaciones mínimo  $l$  cuyas necesidades se han de cubrir. A continuación figura la cantidad de kilos de ayuda necesarios en cada población. Finalmente una fila para cada voluntario indicando la cantidad de kilos de ayuda llevará a cada población.

## Salida

Por cada caso de prueba el programa escribirá una línea con la cantidad de kilos máxima que se puede repartir. En caso de no sea posible cumplir la restricción de cubrir las necesidades de al menos  $l$  poblaciones se escribirá **IMPOSIBLE**.

### Entrada de ejemplo

```
4
1 1 1
3
5
4 3 2
10 7 6
2 3 1
4 3 3
2 1 3
4 4 4
4 3 3
10 7 6
2 3 1
4 3 3
2 1 3
4 4 4
4 3 0
10 7 6
2 3 1
4 3 3
2 1 3
4 4 4
```

### Salida de ejemplo

```
3
13
IMPOSIBLE
14
```