

MATCOM
Segundo examen intrasemestral de Programación
Curso 2023-2024
Parte 1

Juan salado (I)

Después de muchos negocios turbios fallidos y del fiasco del tesoro que resultó ser una única y triste pieza de oro, el infame capitán Juan Gorrión está en la ruina. Las únicas posesiones que mantiene son un pequeño bote, un saco y un escaso capital. Ante esta situación desesperada, decide incursionar en un nuevo negocio turbio (si no es turbio no vale), la venta de... ¿Sal ilegal? La polvorosa sustancia blanca ha resultado tener buena aceptación en la clientela (turbia) del Caribe.

El proceso de comercio de sal ilegal consta de dos fases, la compra y la venta. Primero, en la fase de **compra**, Juan llena su saco (de capacidad máxima n) de forma que se minimice el dinero invertido. Para ello, Juan cuenta con la lista de los k_1 precios de la sal en cada isla que vende esta sustancia, así como la cantidad con la que cuenta cada una. Luego de comprados los bienes, llega la hora de **venderlos**, también, de forma que se maximice el dinero obtenido. Para ello, Juan cuenta con la lista de los k_2 precios de la sal en cada isla dispuesta a comprar, así como la capacidad máxima de sus almacenes. Pero la cosa no acaba aquí (raios), los compradores han decidido que deben llenar sus almacenes por completo o de lo contrario no comprarán nada.

Juan quiere hacer un pequeño estudio de mercado antes de comprar la sal, para así no comprar más sal de la que le será posible vender. Llamaremos **potenciales compradores** a los clientes (que quieran comprar sal) cuya demanda pueda ser satisfecha con la capacidad total del saco. En caso de que la suma de las capacidades de todos los potenciales compradores sea menor que la capacidad del saco (n), Juan solo comprará lo justo para satisfacer sus demandas. Por otro lado, mientras la suma de las capacidades de los potenciales compradores sobrepase n , Juan llenará el saco todo lo que sea posible (comprando todo lo que quede en las capacidades de los vendedores mientras quepa en el saco o llenando el saco si queda suficiente oferta).

Juan hará este proceso una única vez (una compra y una venta).



La capacidad del saco está dada por un entero n . Las listas con (a) los precios de compra, (b) las cantidades de sal que Juan puede comprar, (c) los precios de ventas y (d) las capacidades de los almacenes de compradores se representan con cuatro arrays de enteros *buyPrices*, *buyCapacities*, *sellPrices* y *sellCapacities*, respectivamente. Usted debe implementar el siguiente método:

```
1  public static long Solve(long capacity, long[] buyPrices, long[]  
    buyCapacities, long[] sellPrices, long[] sellCapacities)  
2  {  
3      // Delete this line and code here:  
4      throw new NotImplementedException();  
5  }
```

Debe retornar un entero que representa el balance resultante del proceso de compra y venta más eficiente que puede realizar Juan. Positivo en caso de ganancia y negativo en caso de pérdida.

Aclaraciones:

- Los tamaños de *buyPrices* y *buyCapacities* serán iguales (k_1).

- Los tamaños de *sellPrices* y *sellCapacities* serán iguales (k_2).
- Las capacidades y precios siempre serán enteros no negativos.
- Mientras quede sal disponible a la venta y capacidad en el saco, y no se haya sobrepasado la capacidad de los potenciales compradores, se debe llenar el saco incluso si el precio de compra es tan elevado que el balance final se hace negativo. Juan es un poco incapaz en tema de estudios de mercado.
- Los precios son por unidad.

Ejemplos

ejemplo 1

$$n = 20$$

$$\text{buy prices} = \{4, 2, 1\}$$

$$\text{buy capacities} = \{15, 6, 15\}$$

$$\text{sell prices} = \{3, 3, 5, 50\}$$

$$\text{sell capacities} = \{10, 10, 11, 21\}$$

Primero se compran 15 unidades con valor 1 y 5 con valor 2 ($15 * 1 + 2 * 5 = 25$). Se venden 10 unidades con valor 3 para el primer comprador y otras 10 para el segundo comprador ($10 * 3 + 10 * 3 = 60$). El balance será de 35 ($60 - 25 = 35$).

ejemplo 2

$$n = 20$$

$$\text{buy prices} = \{40, 20, 10\}$$

$$\text{buy capacities} = \{15, 6, 15\}$$

$$\text{sell prices} = \{3, 3, 5, 50\}$$

$$\text{sell capacities} = \{10, 10, 11, 21\}$$

Primero se compran 15 unidades con valor 10 y 5 con valor 20 ($15 * 10 + 20 * 5 = 250$). Se venden 10 unidades con valor 3 para el primer comprador y otras 10 para el segundo comprador ($10 * 3 + 10 * 3 = 60$). El balance será de -190 ($60 - 250 = -190$).



ejemplo 3

$$n = 20$$

$$\text{buy prices} = \{4, 2, 1\}$$

$$\text{buy capacities} = \{15, 6, 15\}$$

$$\text{sell prices} = \{3, 3, 5, 50\}$$

$$\text{sell capacities} = \{2, 2, 11, 21\}$$

Primero se compran 15 unidades con valor 1 ($15 * 1 = 15$ (wao)). Se venden 2 unidades con valor 3 para el primer comprador, otras 2 con valor 3 para el segundo comprador y 11 con valor 5 para el tercer comprador ($2 * 3 + 2 * 3 + 11 * 5 = 67$). El balance será de 52 ($67 - 15 = 52$).

Note que en este caso sólo se compró 15 unidades, lo suficiente para satisfacer a los potenciales compradores.

ejemplo 4

$$n = 20$$

$$\text{buy prices} = \{4, 1, 1\}$$

$$\text{buy capacities} = \{11, 6, 20\}$$

$$\text{sell prices} = \{3, 3, 5, 4\}$$

$$\text{sell capacities} = \{10, 10, 15, 13\}$$

Primero se compran 20 unidades de costo 1 ($20 * 1 = 20$). Se venden 15 unidades al tercer comprador con valor de 5 ($15 * 5 = 75$). El balance será de 75 ($75 - 20 = 55$). El saco queda con 5 unidades.