

A partir de la siguiente página, encontrará el enunciado del problema “cosechadoras”, utilizado recientemente en la Olimpiada Informática Argentina en el Certamen de Selección para la IOI 2026.

En ese problema, es posible demostrar que los números de la salida forman un array de concavidad no positiva. Es decir:

$$ret[i] - ret[i - 1] \geq ret[i + 1] - ret[i] \text{ para } 1 \leq i \leq \lfloor \frac{N}{2} \rfloor - 2$$

Lo que también puede expresarse como la propiedad de “diminishing returns” o “utilidad marginal decreciente”.

¡Tu misión consiste en demostrarlo!

Debes enviar la demostración por correo a competencia.leopoldo.taravilse@gmail.com, indicando el team secret correspondiente. Será corregida y respondida por email, y en caso de ser correcta, se realizará un envío correcto automáticamente.

No se promete responder rápido, pudiendo tardar horas o días, pero todas las demostraciones serán corregidas y procesadas en el orden en que los emails lleguen.

La demostración debe ser algo razonable y legible para un humano. Una demostración excesivamente larga y confusa puede ser rechazada por “Paciencia limit exceeded”.

¡La mejor demostración¹ obtendrá $\frac{1}{2}$ punto adicional!

¹De acuerdo al criterio subjetivo de los organizadores

Comerciendo Cosechadoras

Contribución de Agustín Santiago Gutiérrez

Descripción del problema

Imronbek el uzbeko es un famoso comerciante que se dedica a la compraventa de cosechadoras.

Una cosechadora es un producto industrial gigantesco, sumamente costoso y que requiere un galpón inmenso para guardarse. Es por esto que Imronbek en ningún momento puede tener más de una cosechadora a la vez.

Imronbek quiere analizar la eficiencia de sus decisiones pasadas: registró los precios de mercado p_i de las cosechadoras durante N días consecutivos de la temporada de comercio anterior, $0 \leq i \leq N - 1$.

Los precios se miden en soms, que es la moneda de Uzbekistán. De esta forma, si en el i -ésimo día Imronbek compra una cosechadora paga por ella $p[i]$ soms, y si en el i -ésimo día vende una cosechadora obtiene $p[i]$ soms.

Imronbek inicia la temporada sin cosechadora, debe terminarla sin cosechadora, y en ningún momento puede tener más que una cosechadora al mismo tiempo. Cada día debe hacer exactamente una de las siguientes acciones:

- Vender una cosechadora (debe tener una ese día antes de la venta)
- Comprar una cosechadora (no puede ya tener una ese día antes de la compra)
- Esperar al día siguiente (ni comprar ni vender ese día)

Vender cosechadoras es muy desgastante para Imronbek: le interesa saber para cada entero k entre 0 y $\lfloor \frac{N}{2} \rfloor - 1$ inclusive, cuál es la máxima ganancia neta que puede obtener si vende (y por lo tanto compra) **exactamente** $k + 1$ cosechadoras durante la temporada. Notar que este valor puede ser negativo si vender exactamente $k + 1$ cosechadoras implica necesariamente perder soms.

Descripción de la función

La función `cosechadoras(p)` toma el arreglo de precios y debe retornar un arreglo `ret` de longitud $\lfloor \frac{N}{2} \rfloor$. Para cada entero k tal que $0 \leq k \leq \lfloor \frac{N}{2} \rfloor - 1$, `ret[k]` debe contener la máxima ganancia neta posible al vender exactamente $k + 1$ cosechadoras en la temporada.

Evaluador local

El evaluador local lee de la entrada estándar con el siguiente formato:

- Un entero N
- N enteros $p[i]$

Escribe el resultado de la llamada `cosechadoras(p)`.

Restricciones

$$2 \leq N \leq 200.000$$

$$1 \leq p[i] \leq 10^9$$

Ejemplos

Si la entrada es:

```
6
1 2 3 4 5 6
```

La salida correcta es

```
5 4 3
```

Y si la entrada es

```
7
3 2 1 1 1 1 1
```

La salida correcta es

```
0 0 -1
```

Mientras que la siguiente salida obtendría el 20 % del puntaje (ver sección Puntuación).

```
0 42 -1
```

Puntuación

Si la función retorna un arreglo de la longitud correcta $\lfloor \frac{N}{2} \rfloor$, y están bien calculados el primer y el último elemento (es decir $ret[0]$ y $ret[\lfloor \frac{N}{2} \rfloor - 1]$, donde ret es el arreglo retornado), se obtiene el 20 % del puntaje.

Si además del primer y el último elemento, también está bien calculado el elemento en la posición $\lfloor \frac{N}{4} \rfloor$ (indexando desde 0, es decir el elemento $ret[\lfloor \frac{N}{4} \rfloor]$), se obtiene un 30 % adicional (alcanzando así 50 %).

El 50 % restante se obtiene cuando todo el arreglo ret está correctamente calculado.

Subtareas

1. $N \leq 3$ (4 puntos)
2. $N \leq 4$ (4 puntos)
3. $N \leq 20$ (9 puntos)
4. $N \leq 2000$ (15 puntos)
5. **La salida correcta** es un arreglo ret no decreciente de números no negativos. O sea $0 \leq ret[0] \leq ret[1] \leq \dots \leq ret[\lfloor \frac{N}{2} \rfloor - 1]$ (39 puntos)
6. Sin más restricción (29 puntos)