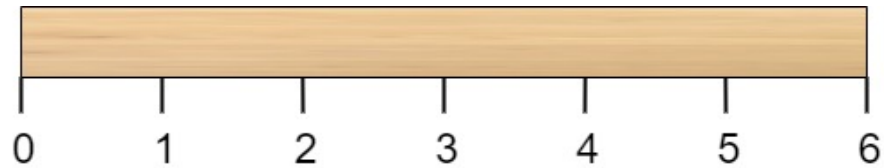


Sea una varilla de madera de longitud n unidades. La varilla está etiquetada desde 0 hasta n . Por ejemplo, una varilla de longitud 6 se etiqueta de la siguiente manera:



Dado un array de enteros **cortes** donde **cortes[i]** indica una posición en la que debe realizarse un corte. Realizar los cortes en algún orden: puede cambiarse el orden de los cortes como se desee.

El costo de un corte es la longitud de la varilla que se va a cortar, el costo total es la suma de los costos de todos los cortes. Cuando se corta una varilla, se dividirá en dos varillas más pequeñas (es decir, la suma de sus longitudes es la longitud de la varilla antes del corte).

Devolver el costo total mínimo de los cortes.

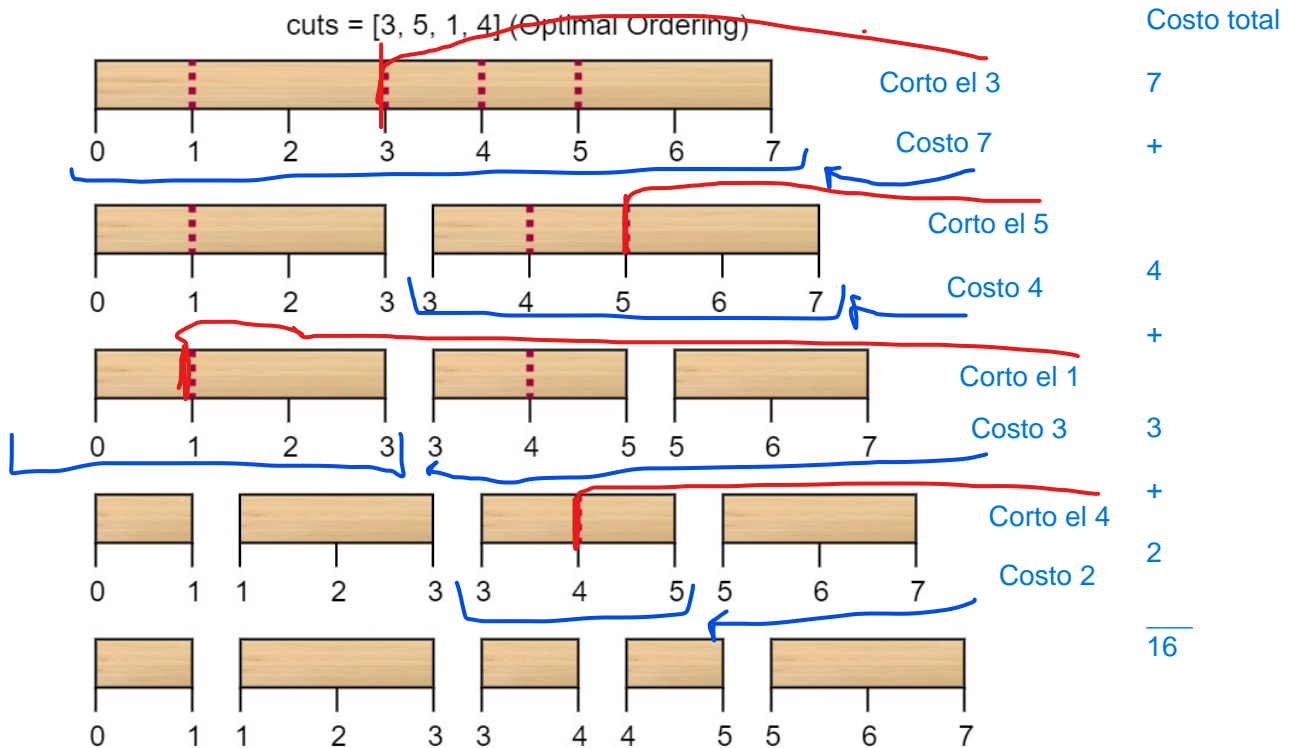
3

$$n * n + 1$$

$$2$$

$$O(n^2)$$

Ejemplo 1



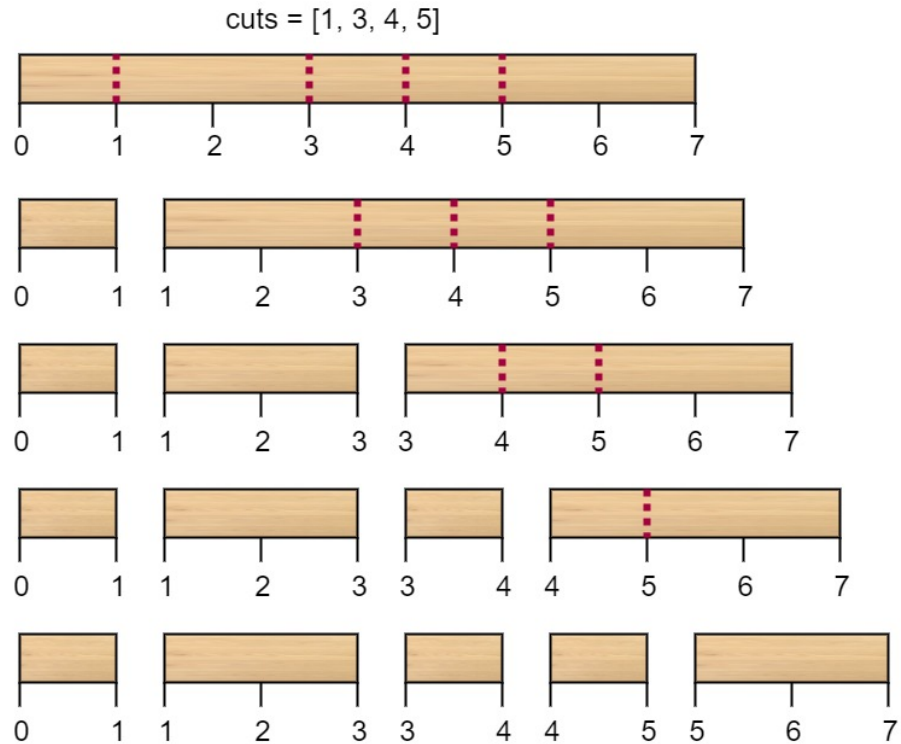
Entrada:

$n = 7$, cortes = [1, 3, 4, 5]

Salida:

16

Explicación: Utilizar el orden de cortes = [1, 3, 4, 5] como en la entrada lleva al siguiente escenario:



El primer corte se realiza en una barra de longitud 7, por lo que el costo es 7. El segundo corte se realiza en una barra de longitud 6 (es decir, la segunda parte del primer corte). El tercero se realiza en una barra de longitud 4. El último corte es en una barra de longitud 3. El costo total es $7 + 6 + 4 + 3 = 20$.

Reorganizar los cortes para que sean [3, 5, 1, 4], por ejemplo, llevará a un escenario con un costo total = 16 (como se muestra en el ejemplo foto $7 + 4 + 3 + 2 = 16$).

Ejemplo 2

Entrada:

$n = 9$, cortes = [5, 6, 1, 4, 2]

Salida: 22

Explicación: Si pruebas los cortes dados en el orden indicado, el costo será 25.

Hay muchos órdenes con costo total ≤ 25 , por ejemplo, el orden $[4, 6, 5, 2, 1]$ tiene un costo total $= 22$ que es el mínimo posible.

Restricciones:

- $2 \leq n \leq 10^6$
- $1 \leq \text{cortes.length} \leq \min(n - 1, 100)$
- $1 \leq \text{cortes}[i] \leq n - 1$
- Todos los enteros en el array cortes son **distintos**.