

beCP

2019

Taak 1.1: Zware dozen (boxes)

Auteur: Robin Jadoul Voorbereiding: Victor Lecomte
Maximale uitvoeringsduur: 1 s Geheugenlimiet: 512 MB

Je werkt in een fabriek en jouw job is om voorwerpen in dozen in te pakken van zodra ze van de productielijn komen. Vandaag zullen er n voorwerpen worden geproduceerd. Het i^{de} voorwerp heeft gewicht w_i . Je hebt k dozen. Je gaat de dozen één voor één vullen met de voorwerpen die komen. Je kan niet meerdere dozen in parallel vullen, dus elke doos zal een aaneensluitende opeenvolging van voorwerpen bevatten: als een doos voorwerp l en voorwerp r bevat, dan moet de doos ook alle voorwerpen tussen l en r bevatten. Elk voorwerp moet in een doos gestoken worden.

Lege dozen zijn heel licht, dus het gewicht van een doos is gewoon de som van de gewichten van de voorwerpen die het bevat. Zware dozen zijn moeilijker om te dragen, dus je wilt *het gewicht van de zwaarste doos minimaliseren*. Wat is het optimale gewicht?

Input

De eerste lijn bevat twee gehele getallen n, k : het aantal voorwerpen en het aantal dozen. De tweede lijn bevat n gehele getallen: de gewichten w_i van de voorwerpen, in de volgorde waarin ze toekomen.

Output

Output een geheel getal op een enkele lijn: het minimale mogelijke gewicht dat de zwaarste doos kan hebben.

Algemene limieten

- $1 \leq n \leq 5 \times 10^5$, het aantal voorwerpen;
- $1 \leq k \leq n$, het aantal dozen;
- $1 \leq w_i \leq 10^{10}$, het gewicht van het i^{de} voorwerp.

Merk op dat de gewichten te groot kunnen zijn op te passen in een 32-bit integer. Gebruik **long long** in C++.

Bijkomende beperkingen

| Subtaak | Punten | Beperkingen |
|---------|--------|-----------------------------|
| A | 10 | $k = 1$ |
| B | 20 | $k = 2$ |
| C | 30 | $n \leq 500$ |
| D | 25 | $n \leq 5000$ |
| E | 15 | Geen bijkomende beperkingen |

Voorbeeld 1

| | |
|----------------------------|------------------|
| sample1.in 3 2 1 4 2 | sample1.out 5 |
|----------------------------|------------------|

In dit voorbeeld heb je 2 dozen. De optimale oplossing steekt de eerste twee voorwerpen in de eerste doos, resulterende in een gewicht van $1 + 4 = 5$, en het derde voorwerp in de tweede doos, resulterende in een gewicht van 2. In dit geval heeft de zwaarste doos een gewicht van 5. Er is geen oplossing waarbij beide dozen een gewicht kleiner dan 5 hebben: merk op dat dozen een aaneensluitende opeenvolging van voorwerpen moeten bevatten.

Deze invoer is geldig voor subtaken B, C en D.

Voorbeeld 2

| | |
|----------------------------|-------------------|
| sample2.in 3 1 2 3 5 | sample2.out 10 |
|----------------------------|-------------------|

In dit voorbeeld heb je maar één doos, dus je moet alle drie voorwerpen erin steken. Het gewicht van die doos zal $2 + 3 + 5 = 10$ zijn.

Deze invoer is geldig voor subtaken A, C en D.

Voorbeeld 3

| | |
|----------------------------------|----------------------------|
| sample3.in 1 1 10000000000 | sample3.out 10000000000 |
|----------------------------------|----------------------------|

Voorbeeld 4

