



## Molekyylien tunnistus

Uolevi on töissä yrityksessä, joka on valmistanut koneen molekyylien tunnistamiseen. Jokaisella molekyyllillä on positiivinen kokonaislukupaino. Koneella on  $tunnistusalue$   $([l, u])$ , missä  $l$  ja  $u$  ovat positiivisia kokonaislukuja. Kone pystyy tunnistamaan joukon molekyyliä tarkalleen silloin, kun joukossa on osajoukko, jonka molekyylien yhteispaino on koneen tunnistusalueen sisällä.

Tarkemmin sanoen  $(n)$  molekyyllillä on positiiviset kokonaislukupainot  $(w_0, \dots, w_{n-1})$ . Tunnistus onnistuu, jos on olemassa indeksijoukko  $(I = \{i_1, \dots, i_m\})$  niin, että  $(l \leq w_{i_1} + \dots + w_{i_m} \leq u)$ .

Koneen rakenteen vuoksi arvojen  $l$  ja  $u$  väli on varmasti suurempi tai sama kuin painavimman ja keveimmän molekyylin kokoero. Tarkemmin sanoen  $(u - l \geq w_{\max} - w_{\min})$ , kun  $(w_{\max} = \max(w_0, \dots, w_{n-1}))$  ja  $(w_{\min} = \min(w_0, \dots, w_{n-1}))$ .

Tehtäväsi on laatia ohjelma, joka joko löytää jonkin molekyylien osajoukon, jonka yhteispaino on tunnistusalueen sisällä, tai havaitsee, että tällaista osajoukkoa ei ole olemassa.

### Toteutuksen yksityiskohdat

Toteuta seuraava funktio:

- `int[] solve(int l, int u, int[] w)`
  - $l$  ja  $u$ : tunnistusalueen rajat,
  - $w$ : molekyylien painot.
  - jos vaadittu osajoukko on olemassa, funktion tulee palauttaa taulukko, jossa on jonkin tällaisen osajoukon molekyylien indeksit. Jos mahdollisia vastauksia on useita, voit palauttaa minkä tahansa niistä.
  - jos vaadittua osajoukkoa ei ole olemassa, funktion tulee palauttaa tyhjä taulukko.

C-kielisen funktion määrittely on hieman erilainen:

- `int solve(int l, int u, int[] w, int n, int[] result)`
  - $n$ : taulukon  $w$  koko (eli molekyylien määrä),
  - muut parametrit ovat samat kuin yllä.
  - funktion ei tule palauttaa taulukkoa vaan kirjoittaa indeksit taulukon `result` ensimmäiseen  $(m)$  kohtaan ja palauttaa  $(m)$ .
  - jos vaadittua osajoukkoa ei ole olemassa, funktion ei tule kirjoittaa mitään taulukkoon `result` ja sen tulee palauttaa  $(0)$ .

Ohjelmasi voi palauttaa indeksit missä tahansa järjestyksessä.

Annetuista koodipohjista selviää, miten toteutus tulee tehdä tarkalleen.

## Esimerkit

### Esimerkki 1

```
solve(15, 17, [6, 8, 8, 7])
```

Tässä esimerkissä on neljä molekyyliä painoilla 6, 8, 8 ja 7. Kone voi tunnistaa molekyylien osajoukkoja, joiden yhteispaino on välillä 15 ja 17. Huomaa, että  $(17-15 \geq 8-6)$ . Molekyylien 1 ja 3 yhteispaino  $(w_1 + w_3 = 8 + 7 = 15)$ , joten funktio voi palauttaa **[1, 3]**. Muut mahdolliset oikeat vastaukset ovat **[1, 2]**  $(w_1 + w_2 = 8 + 8 = 16)$  ja **[2, 3]**  $(w_2 + w_3 = 8 + 7 = 15)$ .

### Esimerkki 2

```
solve(14, 15, [5, 5, 6, 6])
```

Tässä esimerkissä on neljä molekyyliä painoilla 5, 5, 6 ja 6, ja etsittävänä on osajoukko, jonka yhteispaino on välillä 14 ja 15. Huomaa taas, että  $(15-14 \geq 6-5)$ . Ei ole molekyylien osajoukkoa, joiden yhteispaino olisi välillä  $(14)$  ja  $(15)$ , joten funktion tulee palauttaa tyhjä taulukko.

### Esimerkki 3

```
solve(10, 20, [15, 17, 16, 18])
```

Tässä esimerkissä on neljä molekyyliä painoilla 15, 17, 16 ja 18, ja etsittävänä on osajoukko, jonka yhteispaino on välillä 10 ja 20. Huomaa taas, että  $(20-10 \geq 18-15)$ . Mikä tahansa osajoukko, jossa on tarkalleen yksi elementti, täyttää tämän vaatimuksen, joten oikeat vastaukset ovat: **[0]**, **[1]**, **[2]** ja **[3]**.

## Osatehtävät

- (9 pistettä):  $(1 \leq n \leq 100)$ ,  $(1 \leq w_i \leq 100)$ ,  $(1 \leq u, l \leq 1000)$ , kaikki  $(w_i)$  ovat samat.
- (10 pistettä):  $(1 \leq n \leq 100)$ ,  $(1 \leq w_i, u, l \leq 1000)$  ja  $(\max(w_0, \dots, w_{n-1}) - \min(w_0, \dots, w_{n-1}) \leq 1)$ .
- (12 pistettä):  $(1 \leq n \leq 100)$  ja  $(1 \leq w_{i,u,l} \leq 1000)$ .
- (15 pistettä):  $(1 \leq n \leq 10\,000)$  ja  $(1 \leq w_{i,u,l} \leq 10\,000)$ .
- (23 pistettä):  $(1 \leq n \leq 10\,000)$  ja  $(1 \leq w_{i,u,l} \leq 500\,000)$ .
- (31 pistettä):  $(1 \leq n \leq 200\,000)$  ja  $(1 \leq w_{i,u,l} < 2^{\{31\}})$ .

## Esimerkkiarvostelija

Esimerkkiarvostelija lukee syötteen seuraavassa muodossa:

- o rivi 1: kokonaisluvut  $(n)$ ,  $(l)$ ,  $(u)$ .
- o rivi 2:  $(n)$  kokonaislukua:  $(w_0, \dots, w_{n-1})$ .