

## Detecting Molecules

Krešo radi za kompaniju koja proizvodi uređaje za detekciju molekula. Svaka molekula ima težinu koja je prirodan broj. Uređaj ima *raspon detektiranja*  $([l, u])$ , gdje su  $l$  i  $u$  prirodni brojevi. Uređaj može detektirati skup molekula ako i samo ako taj skup sadrži podskup molekula s ukupnom težinom koja je unutar raspona detektiranja.

Formalno, promotrimo  $n$  molekula s težinama  $(w_0, \dots, w_{n-1})$ . Detekcija je uspješna ako postoji skup različitih indeksa  $I = \{i_1, \dots, i_m\}$  takvih da vrijedi  $l \leq w_{i_1} + \dots + w_{i_m} \leq u$ .

Zbog specifičnosti uređaja, garantirano je da je razmak između  $l$  i  $u$  veći ili jednak razlici između najteže i najlakše molekule. Formalno,  $u - l \geq w_{\max} - w_{\min}$ , gdje su  $w_{\max} = \max(w_0, \dots, w_{n-1})$  i  $w_{\min} = \min(w_0, \dots, w_{n-1})$ .

Vaš je zadatak napisati program koji pronalazi bilo koji podskup molekula s ukupnom težinom unutar raspona detekcije, ili zaključuje da ne postoji nijedan takav podskup.

### Implementacijski detalji

Trebate implementirati jednu funkciju (metodu):

- `int[] solve(int l, int u, int[] w)`
  - $l$  i  $u$ : rubovi raspona detekcije,
  - $w$ : težine molekula.
  - ako traženi podskup postoji, funkcija treba vratiti niz indeksa molekula koje čine jedan takav podskup. Ako postoji više točnih rješenja, vratite bilo koje od njih.
  - ako traženi podskup ne postoji, funkcija treba vratiti prazan niz.

Potpis funkcije malo je drugačiji za jezik C:

- `int solve(int l, int u, int[] w, int n, int[] result)`
  - $n$ : broj elemenata niza  $w$  (broj molekula),
  - ostali parametri isti su kao gore.
  - umjesto vraćanja niza od  $m$  indeksa (kao iznad), funkcija treba spremirati indekse u prvih  $m$  polja niza `result` te vratiti  $m$ .
  - ako traženi podskup ne postoji, funkcija ne treba ništa upisati u niz `result` i treba vratiti `0`.

Indekse možete u bilo kojem poretku upisati u povratni niz.

Za implementacijske detalje koristite dane template datoteke.

## Primjer

### Primjer 1

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

U ovom primjeru zadane su četiri molekule s težinama 6, 8, 8 i 7. Uređaj može detektirati podskupove molekula s ukupnom težinom između 15 i 17, uključivo.

Primijetite da je  $(17-15 \geq 8-6)$ . Ukupna težina molekula 1 i 3 je  $(w_1 + w_3 = 8 + 7 = 15)$ , tako da funkcija može vratiti `[1, 3]`. Moguća rješenja su i `[1, 2]` ( $(w_1 + w_2 = 8 + 8 = 16)$ ) i `[2, 3]` ( $(w_2 + w_3 = 8 + 7 = 15)$ ).

### Primjer 2

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

U ovom primjeru zadane su četiri molekule s težinama 5, 5, 6 i 6, te tražimo podskup s ukupnom težinom između 14 i 15, uključivo. Primijetite da je  $(15-14 \geq 6-5)$ . Ne postoji podskup molekula s ukupnom težinom između  $(14)$  i  $(15)$ , stoga funkcija mora vratiti prazan niz.

### Primjer 3

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

U ovom primjeru zadane su četiri molekule s težinama 15, 17, 16 i 18, te tražimo podskup s ukupnom težinom između 10 i 20, uključivo. Primijetite da je  $(20-10 \geq 18-15)$ . Svaki podskup koji se sastoji od jednog člana zadovoljava ograničenja, stoga su točni odgovori: `[0]`, `[1]`, `[2]` i `[3]`.

## Podzadatci

- (9 bodova):  $(1 \leq n \leq 100)$ ,  $(1 \leq w_i \leq 100)$ ,  $(1 \leq u, l \leq 1000)$ , svi  $(w_i)$  su jednaki.
- (10 bodova):  $(1 \leq n \leq 100)$ ,  $(1 \leq w_i, u, l \leq 1000)$ , te  $(\max(w_0, \dots, w_{n-1}) - \min(w_0, \dots, w_{n-1}) \leq 1)$ .
- (12 bodova):  $(1 \leq n \leq 100)$  te  $(1 \leq w_{i,u,l} \leq 1000)$ .
- (15 bodova):  $(1 \leq n \leq 10,000)$  te  $(1 \leq w_{i,u,l} \leq 10,000)$ .
- (23 boda):  $(1 \leq n \leq 10,000)$  te  $(1 \leq w_{i,u,l} \leq 500,000)$ .
- (31 bod):  $(1 \leq n \leq 200,000)$  te  $(1 \leq w_{i,u,l} < 2^{31})$ .

## Priloženi grader

Priloženi grader učitava ulaz u sljedećem formatu:

- linija 1: brojevi  $(n)$ ,  $(l)$ ,  $(u)$ .
- linija 2:  $(n)$  brojeva:  $(w_0, \dots, w_{n-1})$ .