

## Željeznica smrti

Aspazija radi na ringišpilu i za svoju djecu olimpijce planira da napravi novi rolerkoster(roller coaster). Ona je već dizajnirala  $n$  stanica (zgodno numerisanih brojevima od  $0$  do  $n - 1$ ) koje na razne načine utiču na brzinu voza kako bi olimpijcima bio obezbijeđen maksimalni užitak. Sve što je ostalo je da se ove stanice povežu u jednu rutu. Za potrebe ovog zadatka, možete pretpostaviti da je voz zanemarive dužine.

Za svako  $i$  između  $0$  i  $n - 1$ , stanica numerisana brojem  $i$  ima sljedeće dvije osobine:

- kada voz ulazi u ovu stanicu, brzina voza mora biti **manja ili jednaka** od  $s_i$  km/h (kilometara na sat),
- kada voz izlazi iz ove stanice, brzina voza je **tačno**  $t_i$  km/h (nezavisno od brzine kojom je voz ušao u stanicu).

Kako bi završila rolerkoster, Aspazija treba da poveže datih  $n$  stanica u nekom redosljedu koristeći šine između susjednih stanica. Svaka stanica mora biti iskorištena tačno jednom. Aspazija treba da odredi redosljed stanica, a potom i dužinu šina između njih. Dužine šine se mjere u metrima i mogu biti bilo koje nenegativne cjelobrojne dužine (moguće je da budu dužine  $0$ ).

Svaki metar šine između dvije stanice usporava voz za  $1$  km/h. Na početku, voz ulazi u prvu stanicu (određenu redosljedom stanica koji je Aspazija izabrala) brzinom od  $1$  km/h.

Završen rolerkoster mora da zadovoljava sljedeća dva uslova:

- voz ne narušava gornje ograničenje brzine pri ulasku u stanice;
- brzina voza je pozitivna u svakom trenutku.

U svim podzadacima, osim 3. podzadatka, vaš zadatak je da nađete minimalnu moguću ukupnu dužinu šina između stanica. U podzadatku 3 samo trebate provjeriti da li postoji validan redosljed stanica tako da svaka šina ima dužinu nula.

### Detalji implementacije

Potrebno je da implementirate sljedeću funkciju (metodu):

- `int64 plan_roller_coaster(int[] s, int[] t)`
  - $s$ : niz dužine  $n$ , gornja granica za brzinu prilikom ulaska u stanicu.
  - $t$ : niz dužine  $n$ , brzina voza prilikom izlaska iz stanice.
  - Ova funkcija treba da vrati minimalnu moguću ukupnu dužinu šina između

stanica (u 3. podzadatku, funkcija može da vrati bilo koji pozitivan cijeli broj ukoliko je moguće konstruisati rolerkoster, više informacija u objašnjenju podzadatka).

U programskom jeziku C, funkcija treba da ima sljedeći potpis:

- `int64 plan_roller_coaster(int n, int[] s, int[] t)`
  - `n`: broj elemenata u nizovima `s` i `t` (tj., broj stanica),
  - ostali parametri su isti kao ranije.

## Primer

`int64 plan_roller_coaster([1, 4, 5, 6], [7, 3, 8, 6])`

U ovom primeru postoje 4 stanice. Najbolje rješenje je da se stanice posmatraju u redoslijedu `0,3,1,2`, i da se susjedne stanice povežu šinama dužina `1,2,0`, redom. U ovom slučaju voz se kreće na sljedeći način:

- Početna brzina voza je `1` km/h.
- Voz počinje vožnju ulazeći u stanicu numerisanu brojem `0`.
- Voz izlazi iz stanice `0` krećući se brzinom `7` km/h.
- Između stanice `0` i sljedeće stanice postoje šine dužine `1` m. Kada voz stigne do kraja ovih šina, brzina voza je `6` km/h.
- Potom voz ulazi u stanicu numerisanu brojem `3` krećući se brzinom od `6` km/h i izlazi iz nje istom brzinom.
- Poslije izlaska iz stanice numerisane brojem `3`, voz se kreće po šinama dužine `2` m, tako da se brzina voza smanji na `4` km/h pri dolasku do kraja ovog segmenta.
- Po ulasku u stanicu numerisanu brojem `1` voz se kreće brzinom `4` km/h i izlazi iz nje brzinom od `3` km/h.
- Odmah po izlasku iz stanice numerisane brojem `1`, voz ulazi u stanicu numerisanu brojem `2`.
- Na kraju, voz izlazi iz stanice numerisane brojem `2` sa krajnjom brzinom od `8` km/h.

Funkcija treba da vrati ukupnu dužinu šina između stanica:  $1 + 2 + 0 = 3$ .

## Podzadaci

U svim podzadacima važi  $1 \leq s_i \leq 10^9$  i  $1 \leq t_i \leq 10^9$ .

1. (11 poena):  $2 \leq n \leq 8$ ,
2. (23 poena):  $2 \leq n \leq 16$ ,
3. (30 poena):  $2 \leq n \leq 200\,000$ . U ovom podzadatku, vaš program treba samo da provjeri da li je rezultat nula ili ne. Ako rezultat nije nula, bilo koji pozitivni cijeli broj se smatra tačnim rezultatom.
4. (36 points):  $2 \leq n \leq 200\,000$ .

## Opis priloženog gradera

Priloženi grader učitava podatke sa standardnog ulaza u sljedećem formatu:

- linija 1: cijeli broj  $n$ .
- linija  $2 + i$ , za svako  $i$  između  $0$  i  $n - 1$ : cijeli brojevi  $s_i$  i  $t_i$ .