

Roller Coaster Railroad

Krešo radi u lunaparku i planira izgradnju novog rolerkosteru. Već je dizajnirao n stanica (zgodno označenih brojevima od 0 do $n - 1$) koje na razne načine utiču na brzinu vlaka kako bi olimpijcima bio osiguran maksimalni užitak. Preostalo je da se ove stanice povežu u jednu rutu. Za potrebe ovog zadatka, možete pretpostaviti da je vlak zanemarive dužine.

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

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

Kako bi završio rolerkoster, Krešo treba povezati danih n stanica u nekom redosljedu koristeći pruge između susjednih stanica. Svaka stanica mora biti iskorištena točno jednom. Krešo treba odrediti redosljed stanica, a potom i dužinu pruga između njih. Duljine pruga mjere se u metrima i mogu biti bilo koje nenegativne cjelobrojne dužine (moguće je da budu dužine 0).

Svaki metar pruge između dviju stanica usporava vlak za 1 km/h. Na početku, vlak ulazi u prvu stanicu (određenu redosljedom stanica koji je Krešo izabrao) brzinom od 1 km/h.

Završen rolerkoster mora zadovoljiti sljedeća dva uvjeta:

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

U svim podzadacima, osim 3. podzadatka, vaš je zadatak pronaći redosljed stanica i dužine pruga između susjednih stanica tako da su zadovoljeni gornji uvjeti i dodatno je ukupna duljina pruga najmanja moguća. U 3. podzadatku vaš je zadatak provjeriti postoji li ispravan redosljed stanica tako da je svaka pruga duljine 0 .

Implementacijski detalji

Implementirajte sljedeću funkciju (metodu):

- `int64 plan_roller_coaster(int[] s, int[] t)`
 - s : niz duljine n , gornje granice za brzinu prilikom ulaska u stanice.
 - t : niz duljine n , brzine vlaka prilikom izlaska iz stanica.
 - U svim podzadacima osim podzadatka 3, funkcija treba vratiti minimalnu moguću ukupnu duljinu pruga između stanica. U 3. podzadatku, funkcija

treba vratiti 0 ako je moguće konstruirati rolerkoster takav da svaka pruga ima duljinu 0, a inače bilo koji prirodan broj.

U programskom jeziku C, funkcija treba imati 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 isti su kao iznad.

Primjer

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

U ovom primjeru postoje 4 stanice. Najbolje je rješenje stanice izgraditi u redoslijedu `0,3,1,2`, te susjedne stanice povezati prugama duljina `1,2,0`, redom. U ovom slučaju vlak se kreće na sljedeći način:

- Početna je brzina vlaka `1` km/h.
- Vlak počinje vožnju ulazeći u stanicu označenu brojem `0`.
- Vlak izlazi iz stanice `0` krećući se brzinom `7` km/h.
- Između stanice `0` i sljedeće stanice postoje pruge dužine `1` m. Kada vlak stigne do kraja ovih pruga, brzina vlaka je `6` km/h.
- Potom vlak ulazi u stanicu označenu brojem `3` krećući se brzinom od `6` km/h i izlazi iz nje istom brzinom.
- Nakon izlaska iz stanice označene brojem `3`, vlak se kreće po pruzi duljine `2` m, tako da se brzina vlaka smanji na `4` km/h.
- Po ulasku u stanicu označenu brojem `1` vlak se kreće brzinom `4` km/h i izlazi iz nje brzinom od `3` km/h.
- Odmah po izlasku iz stanice označene brojem `1`, vlak ulazi u stanicu označenu brojem `2`.
- Na kraju, vlak izlazi iz stanice označene brojem `2` s krajnjom brzinom od `8` km/h.

Funkcija treba vratjeti ukupnu dužinu pruga između stanica: `1 + 2 + 0 = 3`.

Podzadatci

U svim podzadacima vrijedi $1 \leq s_i \leq 10^9$ te $1 \leq t_i \leq 10^9$.

1. (11 bodova): $2 \leq n \leq 8$,
2. (23 boda): $2 \leq n \leq 16$,
3. (30 bodova): $2 \leq n \leq 200\,000$. U ovom podzadatku, vaš program treba projeriti je li odgovor 0 ili nije; ako nije, bilo koji prirodan broj smatra se točnim odgovorom.
4. (36 bodova): $2 \leq n \leq 200\,000$.

Opis priloženog gradera

Priloženi grader učitava podatke sa standardnog ulaza u sljedećem formatu:

- redak 1: prirodan broj n .
- redak $2 + i$, za svaki i između 0 i $n - 1$: cijeli brojevi s_i, t_i .