



Vuoristorata

Maija on töissä huvipuistossa, ja hänen tehtävänsä on rakentaa uusi vuoristorata. Hän on jo suunnitellut n erikoisosuutta (numeroitu $0 \dots n - 1$), jotka vaikuttavat junan nopeuteen. Hänen täytyy nyt yhdistää kaikki osuudet ja ehdottaa lopullista vuoristorataa. Tässä tehtävässä voit olettaa, että junan pituus on nolla.

Erikoisosuuden i (kun i on $0 \dots n - 1$) ominaisuudet ovat:

- erikoisosuuden alussa on nopeusrajoitus: junan nopeuden tulee olla **enintään** s_i km/h,
- erikoisosuuden päättyessä junan nopeus on **tarkalleen** t_i km/h, riippumatta nopeudesta, jolla juna tuli erikoisosuuteen.

Lopullisen vuoristoradan tulee sisältää jokainen n osuudesta tarkalleen kerran. Lisäksi jokaisen peräkkäisen osuuden välillä tulee olla välirata. Maijan tulee valita n osuuden järjestys ja päättää sen jälkeen kunkin väliradan pituus. Väliradan pituus mitataan metreissä, ja se voi olla mikä tahansa epänegatiivinen kokonaisluku (mahdollisesti nolla).

Jokainen metri välirataa kahden erikoisosuuden välillä hidastaa junaa 1 km/h. Alussa juna aloittaa ensimmäisestä erikoisosuudesta Maijan valitsemassa järjestyksessä nopeudella 1 km/h.

Lopullisen radan tulee täyttää seuraavat vaatimukset:

- juna ei ylitä mitään nopeusrajoitusta, kun se aloittaa erikoisosuuden,
- junan nopeus on positiivinen kaikkina hetkinä.

Kaikissa osatehtävissä paitsi osatehtävässä 3 tehtäväsi on löytää pienin mahdollinen väliratojen yhteispituus. Osatehtävässä 3 sinun riittää tarkistaa, onko olemassa kelvollista vuoristorataa, jossa jokaisen väliradan pituus on nolla.

Toteutuksen yksityiskohdat

Toteuta seuraava funktio:

- `int64 plan_roller_coaster(int[] s, int[] t)`
 - s : taulukko kokoa n , nopeusrajoitukset.
 - t : taulukko kokoa n , loppunopeudet.
 - Funktion tulee palauttaa pienin mahdollinen väliratojen yhteispituus (osatehtävässä 3 voit palauttaa minkä tahansa positiivisen luvun, jos vastaus ei ole nolla).

C-kielessä funktion määrittely on hieman erilainen:

- `int64 plan_roller_coaster(int n, int[] s, int[] t)`
 - `n`: taulukoiden `s` ja `t` koko (eli erikoisosuuksien määrä),
 - muut parametrit samoin kuin yllä.

Esimerkki

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

Tässä esimerkissä on neljä erikoisosuutta. Paras ratkaisu on yhdistää ne järjestyksessä `0, 3, 1, 2`, ja valita väliratojen pituudet `1, 2, 0`. Junan matka kulkee näin:

- Alussa junan nopeus on `1` km/h.
- Juna siirtyy erikoisosuuteen `0`.
- Juna poistuu osuudesta `0` nopeudella `7` km/h.
- Sitten on välirata, jonka pituus on `1` m. Väliradan päätteeksi junan nopeus on `6` km/h.
- Juna siirtyy erikoisosuuteen `3` nopeudella `6` km/h ja poistuu siitä samalla nopeudella.
- Osuuden `3` jälkeen juna kulkee `2` m pituisen väliradan läpi. Sen nopeudeksi tulee `4` km/h.
- Juna siirtyy erikoisosuuteen `1` nopeudella `4` km/h ja poistuu siitä nopeudella `3` km/h.
- Välittömästi erikoisosuuden `1` jälkeen juna siirtyy erikoisosuuteen `2`.
- Juna poistuu osuudesta `2`. Sen loppunopeus on `8` km/h.

Funktion tulee palauttaa väliratojen yhteispituus: $1 + 2 + 0 = 3$.

Osatehtävät

Kaikissa osatehtävissä $1 \leq s_i \leq 10^9$ ja $1 \leq t_i \leq 10^9$.

1. (11 pistettä): $2 \leq n \leq 8$,
2. (23 pistettä): $2 \leq n \leq 16$,
3. (30 pistettä): $2 \leq n \leq 200\,000$. Tässä osatehtävässä ohjelmasi riittää selvittää, onko vastaus nolla vai ei. Jos vastaus ei ole nolla, mikä tahansa positiivinen kokonaisluku tulkitaan oikeaksi.
4. (36 pistettä): $2 \leq n \leq 200\,000$.

Esimerkkisarvostelija

Esimerkkisarvostelija lukee syötteen seuraavassa muodossa:

- rivi 1: kokonaisluku `n`.
- rivi `2 + i` (kun `i` on välillä `0 ... n - 1`): kokonaisluvut `si` ja `ti`.