



Linksmieji kalneliai

Ana dirba atrakcionų parke. Jos užduotis – išdėstyti geležinkelio bėgius naujesiems linksmiesiems kalneliams. Ana jau numatė n traukinuko greitį keičiančių sekcijų, sunumeruotų nuo 0 iki $n - 1$. Jai teliko parinkti šių sekcijų eilės tvarką. Šioje užduotyje laikykite, kad traukinukas neturi ilgio.

Kiekviena sekcija i (i kinta nuo 0 iki $n - 1$) turi du papildomus apribojimus:

- įvažiuojant traukinuko greitis **turi būti mažesnis arba lygus** s_i km/val (kilometrų per valandą);
- išvažiuojant iš sekcijos traukinuko greitis **yra lygus** t_i km/val, nepriklausomai nuo įvažiavimo greičio.

Pabaigti linksmieji kalneliai -- traukinio linija turinti visas n sekcijų, išdėstytų kažkuria tvarka. Kiekviena sekcija šioje linijoje turi būti panaudota lygiai vieną kartą. Iš eilės einančios sekcijos turi būti sujungtos bėgiais. Ana turi pasirinkti šių sekcijų išdėstymo tvarką bei bėgių tarp jų ilgus. Bėgių ilgai yra matuojami metrais ir yra neneigiami sveikieji skaičiai (gali būti lygūs ir nuliui).

Kiekvienas bėgių, jungiančių dvi sekcijas, metras sulėtina traukinuką po 1 km/val. Atrakciono pradžioje traukinukas įriedą į pirmąją sekciją (pirmąją pagal Anos pasirinktą išdėstymą) 1 km/val. greičiu.

Galutinis projektas turi atitikti tokius reikalavimus:

- traukinuko greitis nepažeidžia sekcijų įvažiavimo greičio apribojimo;
- traukinuko greitis visada yra teigiamas.

Visuose dalinėse užduotyse, išskyrus trečiojoje, reikia rasti minimalų bendrą bėgių tarp sekcijų ilgį. Trečiojoje dalinėje užduotyje reikia grąžinti 0 , jei egzistuoja toks sekcijų išdėstymas, kuriame bendras bėgių ilgis būtų lygus nuliui. Priešingu atveju šioje dalinėje užduotyje reikia grąžinti bet kokią teigiamą skaičių.

Realizacija

Realizuokite tokią funkciją (metodą):

- `int64 plan_roller_coaster(int[] s, int[] t)`
 - s : n elementų turintis masyvas, kuris nurodo maksimalius sekcijų įvažiavimo greičius,
 - t : n elementų masyvas, nusakantis išvažiavimo greičius,
 - Ši funkcija turi grąžinti bendrą mažiausią galimą bėgių, jungiančių

specialias sekcijas, ilgį (trečiojoje dalinėje užduotyje galite išvesti bet koki sveikąjį skaičių jei atsakymas nėra nulis – detalesnis paaiškinimas dalinių užduočių aprašo skyriuje).

C kalbai ši funkcija atrodo šiek tiek kitaip:

- `int64 plan_roller_coaster(int n, int[] s, int[] t)`
 - `n`: elementų kiekis masyvuose `s` ir `t` (t.y. specialių sekcijų kiekis),
 - kitų elementų prasmė išlieka tokia pati.

Pavyzdys

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

Šiame pavyzdyje yra keturios sekcijos. Optimaliame sprendinyje šios sekcijos turi būti sujungtos tokia tvarka: `0,3,1,2`. O jas jungiančių bėgių ilgiai atitinkamai turi būti

`1,2,0`. Tuomet traukinukas riedės taip:

- Pradžioje traukinuko greitis bus `1` km/val.
- Šiuo greičiu traukinukas įvažiuos į specialią sekciją nr. `0`.
- Traukinukas `7` km/val. greičiu išvažiuos iš sekcijos nr. `0`.
- Tada traukinukui reikės nuvažiuoti `1` m bėgiais. Pasiekus bėgių galą, traukinukas važiuos `6` km/val greičiu.
- Traukinukas `6` km/val. greičiu įvažiuoja ir išvažiuoja iš specialios sekcijos nr. `3`.
- Po sekcijos nr. `3`, traukinukas važiuos `2` m bėgiais. Taip jo greitis sumažės iki `4` km/val.
- Traukinukas `4` km/val. greičiu įvažiuoja į sekciją nr. `1` ir iš jos išvažiuoja `3` km/val greičiu.
- Vos tik traukinukas palieka sekciją nr. `1`, jis iš karto įvažiuoja į sekciją nr. `2`.
- Galiausiai traukinukas palieka sekciją nr. `2` važiuodamas `8` km/val greičiu.

Funkcija turi grąžinti bendrą bėgių ilgį tarp sekcijų: `1 + 2 + 0 = 3`.

Dalinės užduotys

Visose dalinėse užduotyse galioja: $1 \leq s_i \leq 10^9$ ir $1 \leq t_i \leq 10^9$.

1. (11 taškų): $2 \leq n \leq 8$,
2. (23 taškai): $2 \leq n \leq 16$,
3. (30 taškų): $2 \leq n \leq 200\,000$. Šioje dalinėje užduotyje programa turi patikrinti ar atsakymas bus nulis ar ne. Jei atsakymas nėra nulis, bet koks teigiamas sveikasis skaičius bus laikomas teisingu sprendiniu.
4. (36 taškai): $2 \leq n \leq 200\,000$.

Pavyzdinė vertinimo programa

Pavyzdinė vertinimo programa duomenis skaito tokiu formatu:

- 1-oji eilutė: sveikasis skaičius `n`.

- $(2 + i)$ -oji eilutė: kiekvienam i nuo 0 iki $n - 1$: sveikieji skaičiai s_i ir t_i .