

Augošais sadalījums

Laika ierobežojums: 1,5 sekundes

Atmiņas ierobežojums: 1024 MiB

Boriss un Ihors ir atraduši N pozitīvu veselu skaitļu virkni $a_0, a_1, \dots, a_{N-1}$, un vēlas to sadalīt savā starpā. Viņi nevar vienoties par virknes sadalījumu, tāpēc lūdz par to izlemt jums.

Pirms sadalīšana sākas, jums tiks iedota visa virkne. Pēc tam jūs apstrādājat virknes locekļus no kreisās uz labo pusi: vispirms a_0 , tad a_1 , un tā tālāk līdz a_{N-1} . Katrs virknes loceklis, kas tiek apstrādāts, ir jāpiešķir tieši vienam no abiem – Borisam vai Ihoram.

Viņi abi uzstāj, ka virknes locekļiem, kurus katrs saņem, ir jāveido stingri augoša virkne tādā secībā, kādā locekļi tika saņemti. Tas nozīmē, ka katram loceklim, kas tiek piešķirts kādam no viņiem, ir jābūt stingri lielākam par iepriekšējo locekli, ko saņēma tas pats cilvēks. Pirmais loceklis, ko katrs saņem, var būt ar jebkādu vērtību, un ir iespējams, ka viens no viņiem vispār nesaņem nevienu locekli.

Boriss un Ihors nevar vienoties arī par to, cik locekļu katram no viņiem ir jāsaņem. Līdz ar to jums katram vesalam skaitlim K no 0 līdz N (ieskaitot) ir jāatbild uz vaicājumu: vai ir iespējams sadalīt locekļus tā, lai Boriss saņemtu tieši K locekļus, un pēc sadalīšanas abas izveidotās virknes būtu stingri augošas? Katrai K vērtībai ir neatkarīgs vaicājums, tas ir, visi $N + 1$ sadalīšanas procesi ir pilnībā neatkarīgi cits no cita.

Piemēram, ņemsim virkni $a = [3, 1, 4, 5, 5]$ un aplūkosim gadījumus:

- $K = 3$: jūs varat iedot $a_0 = 3$ Borisam, $a_1 = 1$ Ihoram, $a_2 = 4$ Borisam, $a_3 = 5$ Borisam un $a_4 = 5$ Ihoram. Rezultātā Boriss saņem 3, 4 un 5 tieši šādā secībā, un Ihors saņem 1 un 5 tieši šādā secībā. Abas iegūtās virknes ir stingri augošas un Boriss ir saņēmis 3 locekļus, līdz ar to pie $K = 3$ virkni sadalīt ir iespējams.
- $K = 0$: Boriss nesaņem nevienu locekli un Ihors saņem visus locekļus. Ihora virkne sākas ar 3, 1, tātad viņa virkne nevar būt stingri augoša, tāpēc pie $K = 0$ virkni sadalīt nav iespējams.

Šai virknei vienīgās K vērtības, pie kurām ir iespējams sadalīt virkni, ir 2 un 3.

Implementācijas detaļas

Jums ir jāimplementē šāda funkcija:

```
std::vector<bool> increasing_split(std::vector<int> a)
```

- a : masīvs garumā N , dotā skaitļu virkne;

Šai funkcijai ir jāatgriež `bool` masīvs garumā $N + 1$. Masīva i -tajam elementam ir jābūt `true`, ja virknes locekļus var sadalīt tā, lai Boriss saņemtu tieši i locekļus un abas izveidotās virknes būtu stingri augošas, un `false` pretējā gadījumā. Šī funkcija tiek izsaukta tieši vienu reizi katrā testā.

Ierobežojumi

- $2 \leq N \leq 4 \cdot 10^5$;
- $1 \leq a_i \leq 10^9$ visiem $0 \leq i < N$.

Piemēri

Piemēra ievaddati	Piemēra izvaddati
5 3 1 4 5 5	001100
4 1 2 3 4	11111

1. piemēra izskaidrojums

Dota virkne $a = [3, 1, 4, 5, 5]$, tā pati virkne, kas tiek izmantota uzdevuma aprakstā. Apskatām katru K vērtību:

- $K = 0$: kā norādīts uzdevuma aprakstā, neeksistē derīgs sadalījums, tāpēc atbilde ir 0;
- $K = 1$: līdzīgi var pierādīt, ka neeksistē derīgs sadalījums, kurā Boriss saņem tieši vienu locekli, tāpēc atbilde ir 0;
- $K = 2$: virkni var sadalīt šādi: $a_1 = 1$ un $a_4 = 5$ Borisam, $a_0 = 3$, $a_2 = 4$ un $a_3 = 5$ Ihoram. Boriss saņem 1 un 5 tieši šādā secībā, Ihors saņem 3, 4 un 5 tieši šādā secībā; abas virknes ir stingri augošas, tāpēc atbilde ir 1;
- $K = 3$: kā norādīts uzdevuma aprakstā, eksistē derīgs sadalījums, tāpēc atbilde ir 1;
- $K = 4$ un $K = 5$: neeksistē derīgs sadalījums, tāpēc atbilde ir 0.

2. piemēra izskaidrojums

Dota virkne $a = [1, 2, 3, 4]$, kas jau ir stingri augoša. Līdz ar to, neatkarīgi no tā, kā virknes locekļi tiek sadalīti starp Borisu un Ihoru, katrs no viņiem saņems stingri augošu virkni. Līdz ar to katrai K vērtībai no 0 līdz 4 ir derīgs sadalījums.

Apakšuzdevumi

Apakšuzdevums	Punkti	N	Papildu ierobežojumi
0	0	-	Piemēri.
1	10	$N \leq 18$	-
2	5	$N \leq 4 \cdot 10^5$	$a_i \leq a_{i+1}$ visiem $0 \leq i < N - 1$.
3	5	$N \leq 4 \cdot 10^5$	$a_0 \geq \max(a_1, a_2, \dots, a_{N-1})$.
4	16	$N \leq 4 \cdot 10^5$	a ir permutācija skaitļu virknei $1, 2, \dots, N$. Katram indeksam $0 \leq i < N - 1$, kuram izpildās $a_i < a_{i+1}$, virknes prefikss $a_0, a_1, \dots, a_i$ ir permutācija skaitļiem $1, 2, \dots, i + 1$.
5	21	$N \leq 5\,000$	a ir permutācija skaitļu virknei $1, 2, \dots, N$. Katram indeksam $0 \leq i < N - 1$, kuram izpildās $\max(a_0, a_1, \dots, a_i) < a_{i+1}$, virknes prefikss $a_0, a_1, \dots, a_i$ ir permutācija skaitļiem $1, 2, \dots, i + 1$.
6	17	$N \leq 4 \cdot 10^5$	a ir permutācija skaitļu virknei $1, 2, \dots, N$. Katram indeksam $0 \leq i < N - 1$, kuram izpildās $\max(a_0, a_1, \dots, a_i) < a_{i+1}$, virknes prefikss $a_0, a_1, \dots, a_i$ ir permutācija skaitļiem $1, 2, \dots, i + 1$.
7	16	$N \leq 5\,000$	-
8	10	$N \leq 4 \cdot 10^5$	-

Piemēra vērtētājprogramma

Piemēra vērtētājprogramma nolasa ievadi šādā formātā:

- 1. rinda: viens vesels skaitlis – N vērtība;
- 2. rinda: N pozitīvi veseli skaitļi $a_0, a_1, \dots, a_{N-1}$ – virknes locekļi.

Ja atgrieztais masīvs nav garumā $N + 1$, piemēra vērtētājprogramma atgriezīs verdiktu `WA: Returned array does not have size N+1`. Pretējā gadījumā, vērtētājprogramma izvadīs rezultātu šādā formātā:

- 1. rinda: ciparu virkne garumā $N + 1$, kas sastāv no cipariem 0 un 1. Virknes i -tais cipars ir 1, ja pie $K = i$ eksistē derīgs sadalījums, un 0 pretējā gadījumā.