

D. Surašymas (census)

Laiko apribojimas: 1 sekundė

Atminties apribojimas: 128 MiB

Mažiau žinomas faktas apie Cesenatico miestelį yra tas, kad čia įsikūrusi slapta N informatikų draugija. Ši draugija yra ištis labai slapta; jokia narė nepažįsta kitų narių. Kiekviena narė turi unikalų ID: neneigiamą sveikąjį skaičių I .

Vienintelis bendravimas tarp narių yra netiesioginis, naudojant kreida užrašytus skaičius įvairiose miesto vietose. Kas 100 metų draugija atlieka surašymą, kad suskaičiuotų savo nares. Surašymui pasibaigus, kiekviena narė turėtų žinoti bendrą draugijos narių skaičių.

Surašymas vyksta kelias dienas. Kiekvieną dieną kiekviena narė, vis dar dalyvaujanti šiame procese, pasirenka ir atlieka lygiai vieną veiksmą: **skaityti**, **rašyti** arba **sustoti** dalyvauti.

- Jei narė pasirenka **skaityti**, ji pasirenka vietą P . Dienos metu narė apsilanko vietoje P ir perskaito ten parašytą skaičių.
- Jei narė pasirenka **rašyti**, ji pasirenka vietą P ir skaičių V . Vėlai vakare narė apsilanko vietoje P ir pakeičia ten buvusį skaičių į V . Kadangi jau tamsu, prieš įrašant naują skaičių, senojo perskaityti neįmanoma.
- Jei narė pasirenka **sustoti**, ji nebeatlieka jokių veiksmų kitomis dienomis.

Jei viena narė pamato kitą narę, rašančią skaičių, ji gali šią narę atpažinti. Todėl griežtai draudžiama dviem ar daugiau narių tą pačią dieną rašyti toje pačioje vietoje. (Skaitymui tokių apribojimų nėra, nes tai galima atlikti nepastebimai.)

Jei viena ar daugiau narių skaito iš vietos, kurioje kita narė tą pačią dieną nori rašyti, visi skaitymai įvyksta prieš rašymą.

Kaip draugija turėtų planuoti savo surašymo procesą, kad sumažintų dienų skaičių, kol visos sužinos teisingą narių skaičių?

Įgyvendinimas

⇒ Tai interaktyvi užduotis, kurioje vienu metu bus vykdomas nežinomas skaičius ($1 \leq N \leq 100$) jūsų programos vienetų. Kiekvienas programos vienetas imituoja vieną draugijos narę.

Yra 10^{18} vietų. Vietos numeris P turi tenkinti sąlygą $0 \leq P < 10^{18}$. Iš pradžių visose vietose įrašyta reikšmė $V = 0$.

Nauja reikšmė V , įrašoma vietoje, visada turi būti sveikasis skaičius $0 \leq V \leq 10^9$. Daugumoje testų grupių V gali būti tik 0 arba 1. Daugiau informacijos rasi Vertinimo skiltyje.

Kai paleidžiamas tavo programos vienetas, jis pirmiausia turi perskaityti eilutę su dviem sveikaisiais skaičiais I ir M ($0 \leq I \leq M - 1$): šio programos vieneto atstovaujamos draugijos narės unikalų ID ir bendrą galimų ID skaičių. Kiekviename testavimo atvejuje visi programos vienetai gaus tą pačią M reikšmę ir skirtingas I reikšmes. Atkreipk dėmesį, kad gali būti ID, kurie nėra priskirti jokiui nariui.

Tada, kiekvieną surašymo proceso dieną, tavo programa turėtų pasirinkti norimą atlikti veiksmą ir atitinkamai išvesti eilutę:

Veiksmas	Reikšmė
$r\ P$	Skaityti vietą P . Po šios eilutės išvedimo tavo programa turi perskaityti eilutę su dabartine vietoje, kurios numeris P , įrašyta reikšme.
$w\ P\ V$	Rašyti į vietą P naują reikšmę V . Jei keli programos vienetai rašo toje pačioje vietoje, kurios numeris P , tą pačią dieną, tavo programa gaus įvertinimą <i>Netinkamas</i> (<i>Not correct</i>). Išskyrus pavyzdžius ir 3-ą testų grupę, turi įrašyti $0 \leq V \leq 1$; žr. Vertinimo skiltį.
$!\ N$	Atsakyti ir sustoti : pranešti, kad yra N narių ir nustoti dalyvauti surašyme. Po atsakymo tavo programa turi įprastai užsibaigti . (Atkreipk dėmesį, kad kiti tavo programos vienetai gali tęsti darbą papildomas dienas, kol jie atsakys ir užsibaigs.)

Jei bet kuris tavo programos vienetas atsako neteisingą N reikšmę, pažeidžia protokolą, naudoja daugiau nei 500 dienų arba viršija (proceso) laiko/atminties limitą, tavo sprendimas bus įvertintas kaip *Netinkamas* (*Not correct*) tame testavimo variante.

Priešingu atveju tavo programa bus (*Iš dalies*) *Teisinga* (*Partially Correct*) tame testavimo atvejuje ir įvertinta pagal reikšmę D - didžiausią dienų skaičių, kurį bet kuris programos vienetas sunaudojo atsakymui. Norėdama gauti maksimalų balų skaičių, turi išspręsti kiekvieną testavimo atvejį su $D \leq 61$ ir $V \leq 1$. Detales rasi Vertinimo skiltyje.

Flushing procedūra. Jei nenaudoji pateiktų šablonų, būtinai atlik procedūrą **flush** po kiekvienos išvestos eilutės, kitaip tavo programa gali būti įvertinta kaip *Netinkamas* (*Not correct*). Python kalboje **flush** procedūra vyksta automatiškai, jei eilutėms skaityti naudoji `input()`. C++ kalboje `cout << endl`; ne tik išveda naują eilutę, bet ir atlieka **flush** procedūrą; bet jei naudoji `printf`, tai naudoki `fflush(stdout)`.

Apribojimai

- $1 \leq N \leq 100$;
- $1 \leq M \leq 100\,000$;
- Gali sunaudoti daugiausiai 500 dienų.

Vertinimas

Tavo programa bus tikrinama su keliais testavimo atvejais, suskirstytais į testų grupes. Norėdama gauti taškus už testų grupę, turi teisingai išspręsti visus jos testavimo atvejus.

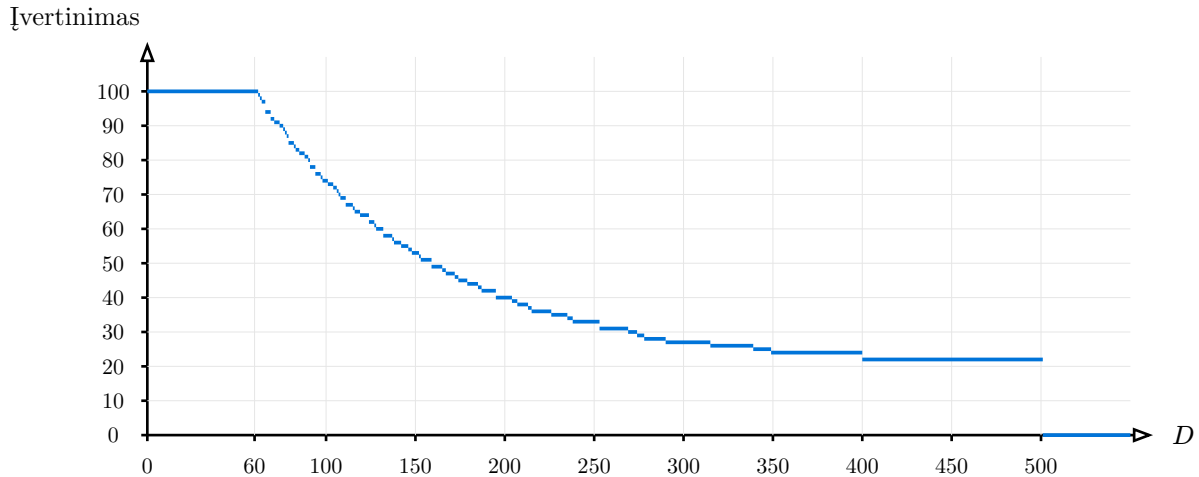
- **0-a testų grupė [0 taškų]**: Pavyzdžiai (gali rašyti bet kokį sveikąjį skaičių $0 \leq V \leq 1\,000\,000\,000$).
- **1-a testų grupė [11 taškų]**: $M \leq 100$, ir N narių turi ID $0, 1, \dots, N - 1$.
- **2-a testų grupė [12 taškų]**: $1 \leq N \leq 2$.
- **3-a testų grupė [22 taškai]**: $M \leq 8000$, ir gali rašyti bet kokį sveikąjį skaičių $0 \leq V \leq 1\,000\,000\,000$.
- **4-a testų grupė [55 taškai]**: Jokių papildomų apribojimų.

1-oje, 2-oje ir 4-oje testų grupėse, atlikdama rašymo veiksmą, gali rašyti tik $V = 0$ arba $V = 1$.

Tegu X_s yra maksimalūs taškai už testų grupę s (pateiktos aukščiau), o D_s yra didžiausias dienų skaičius, kurį bet kuris tavo programos vienetas panaudoja testavimo atvejuje iš testų grupės s . Tada:

$$\text{score}_s = \begin{cases} X_s & \text{jei } D_s \leq 61 \\ X_s \cdot (0.2 + 0.8 \cdot 1.01^{(60-D_s)}) & \text{jei } 61 < D_s \leq 500 \\ 0 & \text{jei } 500 < D_s. \end{cases}$$

Įvertinimo reikšmė score_s apvalinama iki artimiausio sveikąjį skaičių kiekvienai testų grupei, o tavo bendras rezultatas yra šių balų suma. Norėdama gauti maksimalų balų skaičių už užduotį, kiekviename teste turi pasiekti $D \leq 61$ ir $V \leq 1$.



Pav. 1: Bendras rezultatas, darant prielaidą, kad kiekviena testų grupė išspręsta su ta pačia maksimalia D reikšme.

Pavyzdžiai

Pirmas pavyzdys. Kiekviena stulpelių pora rodo komunikaciją tarp vertintojo ir vieno programos vieneto.

Ver.	Vien.	Ver.	Vien.	Ver.	Vien.	Ver.	Vien.	Ver.	Vien.
	0		1		2		3		4
0 100		1 100		2 100		3 100		4 100	
	w 12 1		w 50 1		w 99 0		w 7 1		r 5
								0	
	r 50		r 7		r 12		w 1 1		! 5
1		1		1					
	! 5		r 1		w 0 0		! 5		
		1							
		! 5			! 5				

Antras pavyzdys.

Vertintojas	Vienetas 0
0 8000	
	w 0 0
	w 1 1
	r 2
1	
	! 2

Vertintojas	Vienetas 1
3 8000	
	w 2 1
	r 1
0	
	r 2
1	
	r 1
1	
	! 2

Paiškinimas

Pirmas pavyzdys. Turime $N = 5$ nars su iš eilės einančiais ID 0, 1, 2, 3, 4 ir $M = 100$ (tinka 1, 3 ir 4 testų grupėms). Programos vienetas i atitinka narę su ID i . Aukščiau pateikta sąveika yra tik viena iš galimų taisyklės atitinkančių veiksmų sekų ir nėra skirta parodyti efektyvią ar protingą strategiją; ji rodoma tik tam, kad būtų iliustruota, kaip veikia protokolas.

Antras pavyzdys. Turime $N = 2$ nars, su ID 0 ir 3, bei $M = 8000$ (tinka 2, 3 ir 4 testų grupėms). Pirmą dieną narė, kurios ID yra 0, įrašo 0-ą vietoje, kurios numeris 0 (jokių pokyčių), o narė, kurios ID yra 3, įrašo 1-ą vietoje, kurios numeris 2.

Vieta	0	1	2	3	4	...
Skaičius	0	0	1	0	0	...

Antrą dieną narė, kurios ID 0, įrašo 1-ą vietoje, kurios numeris 1, o narė, kurios ID 3, skaito iš tos pačios vietos. Atkreipk dėmesį, kad skaitymas vyksta dienos metu, prieš rašymą vakare. Todėl narė, kurios ID 3, vis dar mato 0.

Vieta	0	1	2	3	4	...
Skaičius	0	1	1	0	0	...

Trečią dieną jos abi skaito vietą, kurios numeris 2, ir šioje vietoje įrašytas 1.

Ketvirtą dieną narė, kurios ID 0, atsako, kad yra 2 narės (teisingai), o narė, kurios ID 3, skaito 1-ą iš vietos, kurios numeris 1. Narė, kurios ID 0, iškart po to sustoja ir ateinančiomis dienomis nebedalyvauja.

Galiausiai, dieną $D = 5$, likusi narė taip pat teisingai atsako $N = 2$.

Testavimas

Kad būtų lengviau išbandyti sprendimą, pateikiame paprastą įrankį, kurį gali atsisiųsti iš CMS. Įrankiu naudotis neprivaloma. Atkreipk dėmesį, kad oficiali vertinimo programa CMS sistemoje skiriasi nuo šio testavimo įrankio.

Norint naudoti įrankį, tau reikės pradinių duomenų failo. Gali naudoti pateiktus pavyzdžius `census.input0.txt` ir `census.input1.txt` arba susikurti savo. Pradinių duomenų failas turi prasi-dėti narių skaičiumi N ir galimais ID M . Kitoje eilutėje turi būti pateikta su N skaičių, nurodančių draugijos narių ID.

Python programai, tarkime `census.py` (paprastai paleidžiamai su `pypy3 census.py`), testavimo įrankį paleisk taip:

```
python3 testing_tool.py pypy3 census.py < census.input0.txt
```

C++ programai pirmiausia sukompiliuok savo sprendimą:

```
g++ -DEVAL -std=gnu++20 -O2 -pipe -static -s -o census census.cpp
```

ir tada paleisk testavimo įrankį:

```
python3 testing_tool.py ./census < census.input0.txt
```

Atkreipk dėmesį, kad šioje užduotyje standartinė išvestis (angl. standard output) naudojama bendravimui su vertinimo programa, todėl ji neturėtų būti naudojama programos veikimo derinimui (debugging). Vietoj to gali naudoti standartinę klaidų išvestį (stderr). C++ kalboje gali naudoti `cerr` << `msg` << `endl`; Python kalboje gali naudoti `print(msg, file=sys.stderr)`.

Testavimo įrankis perskaitys ir pateiks šias stderr žinutes kartu su visų tavo programos vienetų atliktomis užklausomis. Atkreipk dėmesį, kad dėl techninių priežasčių jos gali pasirodyti šiek tiek nesinchronizuotos tarpusavyje.