

D. Manntal (census)

Tímamörk: 1 sekúndur

Minnismörk: 128 MiB

Það er minna þekkt staðreynd um Cesenatico að þar er leynifélag N kvenkyns tölvunarfræðinga. Þetta félag er afar leynilegt því enginn meðlimur þekkir neinn annan. Hver meðlimur hefur einkvæmt kennimerki: ekki-neikvæða heiltölu I .

Einu samskiptin milli meðlima eru óbein, í gegnum tölur krassaðar með krít á ýmsum stöðum víðsvegar um bæinn. Á 100 ára fresti framkvæmir félagið manntal til að telja meðlimina. Eftir að manntalinu lýkur ættu allir meðlimir að vita heildarfjölda meðlima í félaginu.

Manntalið stendur yfir í nokkra daga. Á hverjum degi mun hver meðlimur sem enn tekur þátt í ferlinu velja og framkvæma nákvæmlega eina aðgerð: að **lesa**, að **skrifa**, eða að **hætta** þátttöku.

- Ef meðlimur velur að **lesa**, velur hún staðsetningu P . Yfir daginn heimsækir hún staðsetningu P og les töluna sem er skrifuð þar.
- Ef meðlimur velur að **skrifa**, velur hún staðsetningu P og tölu V . Seint um kvöldið heimsækir hún staðsetningu P og breytir tölunni sem var þar í V . Þar sem það er orðið dimmt getur hún ekki lesið gömlu töluna áður en hún skrifar þá nýju.
- Ef meðlimur velur að **hætta**, framkvæmir hún engar aðgerðir næstu daga.

Ef einn meðlimur sér annan skrifa tölu gæti hún þekkt hana. Þess vegna er stranglega bannað að tveir eða fleiri meðlimir velji að skrifa á sömu staðsetningu á sama degi. Það er engin slík takmörkun á því að lesa, þar sem það er hægt að gera í laumi.

Ef einn eða fleiri meðlimir lesa af staðsetningu þar sem annar meðlimur ætlar að skrifa á sama degi, þá gerist allur lestur áður en skrifað er.

Hvernig ætti félagið að skipuleggja manntalið til að lágmarka fjölda daga þar til allir vita réttan fjölda meðlima?

Útfærsla



Þetta er gagnvirkur verkefni, þar sem óþekktur fjöldi tilvika ($1 \leq N \leq 100$) af forritinu þínu verður keyrður samtímis. Hvert tilvik hermí eftir einum meðlimi félagsins.

Það eru 10^{18} staðsetningar. Númer staðsetningar P verður að uppfylla $0 \leq P < 10^{18}$. Í upphafi er gildið á öllum stöðum $V = 0$.

Nýja gildið V sem er skrifað á staðsetningu verður alltaf að vera heiltala þannig að $0 \leq V \leq 10^9$. Í flestum hlutverkefnum getur V aðeins verið 0 eða 1. Sjá stigagjöf fyrir nánari upplýsingar.

Þegar tilvik af forritinu þínu byrjar, ætti það fyrst að lesa línu með tveimur heiltölum, I og M ($0 \leq I \leq M - 1$): einkvæmt kennimerki meðlimsins sem þetta tilvik stendur fyrir og heildarfjölda mögulegra kennimerkja. Innan hvers prófunartilviks fá öll tilvik sama gildi á M og mismunandi gildi á I . Athugaðu að það gætu verið kennimerki sem enginn meðlimur hefur.

Síðan, fyrir hvern dag í manntalinu, ætti forritið að velja þá aðgerð sem það vill framkvæma og prenta línu í samræmi við það:

Aðgerð	Merking
$r \ P$	Les a staðsetningu P . Eftir að prenta þessa línu ætti forritið að lesa línu með núverandi gildi á P .
$w \ P \ V$	Skrifa á stað P nýja gildið V . Ef mörg tilvik skrifa á sama P á sama degi færðu dóminn <i>Rangt</i> . Nema í sýnidæmunum og hlutverkefni 3 verður þú að skrifa $0 \leq V \leq 1$; sjá stigagjöf.
$! \ N$	Svara og hætta : tilkynna að það séu N meðlimir og hætta þátttöku í manntalinu. Eftir að hafa svarað ætti forritið að hætta keyrslu á venjulegan hátt . (Athugaðu að önnur tilvik af forritinu gætu haldið áfram að keyra í fleiri daga áður en þau svara og hætta keyrslu.)

Ef eitthvert tilvik af forritinu þínu svarar með röngu gildi á N , brýtur reglurnar, notar meira en 500 daga, eða fer yfir tíma- eða minnismörk (fyrir hvert ferli), verður lausnin þín dæmd sem *Rangt* fyrir viðkomandi prófunartilvik.

Annars verður forritið þitt dæmt (*Að hluta til*) *Rétt* í prófunartilvikinu og fær stig út frá gildinu D : hámarksfjöldi daga sem eitthvert tilvik var að svara. Til að fá fullt hús stiga þarftu að leysa hvert prófunartilvik með $D \leq 61$ og $V \leq 1$. Sjá stigagjöf fyrir nánari upplýsingar.

Tæma frágag (flush). Ef þú notar ekki sniðmátin sem fylgja með, passaðu þá að tæma staðalfrágag (flush) eftir að þú prentar hverja línu, annars gæti forritið þitt verið dæmt sem *Rangt*. Í Python gerist þetta sjálfkrafa ef þú notar `input()` til að lesa línur. Í C++ tæmir `cout << endl`; frágagið auk þess að prenta nýlínu; ef þú notar `printf`, notaðu `fflush(stdout)`.

Takmarkanir

- $1 \leq N \leq 100$.
- $1 \leq M \leq 100\,000$.
- Þú mátt í mesta lagi nota 500 daga.

Stigagjöf

Forritið þitt verður prófað með nokkrum prófunartilvikum sem er skipt niður í hlutverkefni. Til að fá stig fyrir hlutverkefni þarftu að leysa öll próf í því rétt.

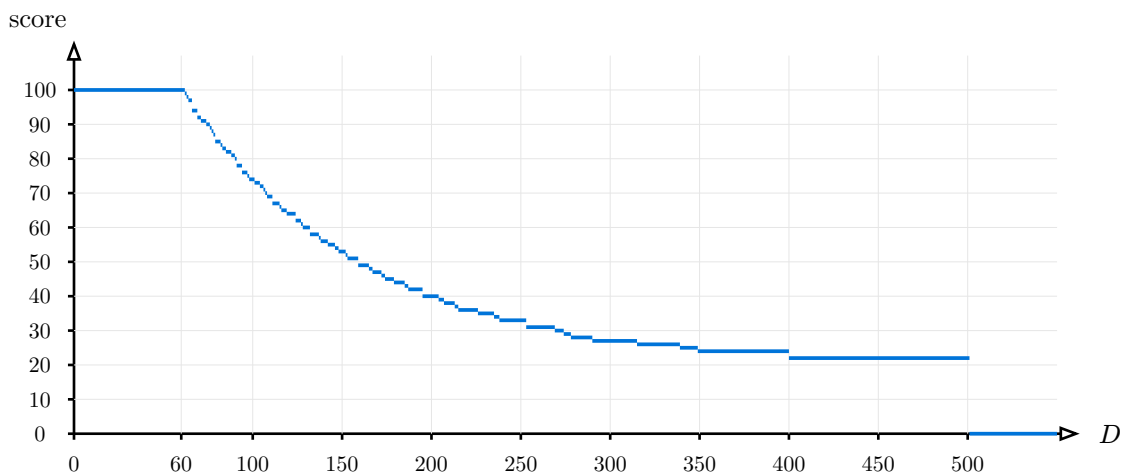
- **Hlutverkefni 0 [0 stig]**: Sýnidæmi (þú mátt skrifa hvaða heiltölu sem er $0 \leq V \leq 1\,000\,000\,000$).
- **Hlutverkefni 1 [11 stig]**: $M \leq 100$, og N meðlimirnir hafa kennimerkin $0, 1, \dots, N - 1$.
- **Hlutverkefni 2 [12 stig]**: $1 \leq N \leq 2$.
- **Hlutverkefni 3 [22 stig]**: $M \leq 8000$, og þú mátt skrifa hvaða heiltölu sem er $0 \leq V \leq 1\,000\,000\,000$.
- **Hlutverkefni 4 [55 stig]**: Engar viðbótartakmarkanir.

Í hlutverkefnum 1, 2 og 4 mátt þú aðeins skrifa $V = 0$ eða $V = 1$ í hverri skrifa-aðgerð.

Látum X_s vera hámarksstig fyrir hlutverkefni s (sýnt hér að ofan), og D_s vera mesta fjölda daga sem eitthvert forritið þitt notar á prófi í hlutverkefni s . Þá:

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

Gildið á stig_s er námundað að næstu heiltölu fyrir hvert hlutverkefni, og heildarstigin þín eru summan af þeim. Til að fá fullt hús stiga fyrir verkefnið þarftu $D \leq 61$ og $V \leq 1$ á hverju prófunartilviki.



Mynd 1: Heildarstig, að því gefnu að hvert hlutverkefni sé leyst með sama hámarki á D .

Dæmi

Fyrsta sýnidæmið. Hvert dálkavar sýnir samskiptin á milli yfirferðarforrits og eins tilviks.

Yfirf.	Tilvik	Yfirf.	Tilvik	Yfirf.	Tilvik	Yfirf.	Tilvik	Yfirf.	Tilvik
0	0	1	1	2	2	3	3	4	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				

Annað sýnidæmið.

Yfirferðarforrit	Tilvik 0	Yfirferðarforrit	Tilvik 1
0 8000		3 8000	
	w 0 0		w 2 1
			r 1
	w 1 1		
		0	
	r 2		r 2
1		1	
	! 2		r 1
		1	
			! 2

Útskýring

Fyrsta sýnidæmið. Við erum með $N = 5$ meðlimi með samliggjandi kennimerki 0, 1, 2, 3, 4 og $M = 100$ (gilt fyrir hlutverkefni 1, 3 og 4). Tilvik i samsvarar meðlimnum sem hefur kennimerkið i . Samskiptin hér að ofan eru bara ein möguleg lögleg röð aðgerða og eru **ekki** hugsuð sem skilvirk eða skynsöm aðferð; hún er aðeins sýnd til að útskýra hvernig samskiptin virka.

Annað sýnidæmið. Við erum með $N = 2$ meðlimi, með kennimerkin 0 og 3, og $M = 8000$ (gilt fyrir hlutverkefni 2, 3 og 4). Á fyrsta degi skrifar meðlimur með kennimerki 0 töluna 0 á stað 0 (engin breyting), og meðlimur með kennimerki 3 skrifar 1 á stað 2.

location	0	1	2	3	4	...
value	0	0	1	0	0	...

Á öðrum degi skrifar kennimerki 0 töluna 1 á stað 1, og kennimerki 3 les sama stað. Taktu eftir að lesturinn gerist yfir daginn, áður en skrifað er um kvöldið. Þess vegna sér kennimerki 3 ennþá 0.

location	0	1	2	3	4	...
value	0	1	1	0	0	...

Á þriðja degi lesa þau bæði stað 2, þar sem 1 er skrifað.

Á fjórða degi svarar kennimerki 0 að það séu 2 meðlimir (rétt), á meðan kennimerki 3 les töluna 1 á stað 1. Kennimerki 0 hættir strax eftir þetta og tekur ekki þátt næstu daga.

Að lokum, á degi $D = 5$, svarar eftirstandandi meðlimurinn líka rétt að $N = 2$.

Prófanir

Til að auðvelda þér að prófa lausnina þína bjóðum við upp á einfalt tól sem þú getur sótt af CMS. Það er valfrjálst að nota tólið. Athugaðu að opinbera yfirferðarforritið á CMS er annað en prófunartólið.

Til að nota tólið þarftu inntaksskrá. Þú getur notað sýnidæmin `census.input0.txt` og `census.input1.txt` sem fylgja með, eða búið til þína eigin skrá. Inntaksskráin ætti að byrja á fjölda meðlima N og mögulegum auðkennum M , á eftir komi lína með N tölum sem tilgreina kennimerkja meðlima félagsins.

Fyrir Python forrit, til dæmis `census.py` (venjulega keyrt með `pypy3 census.py`) keyrirðu prófunartólið svona:

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

Fyrir C++ forrit, þýddu fyrst lausnina þína:

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

og keyrðu síðan prófunartólið:

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

Athugaðu að í þessu verkefni er staðalfrálag notað til að eiga samskipti við yfirferðarforritið, þannig að það ætti ekki að nota það fyrir aflúsun. Í staðinn geturðu notað staðalvilluúttak (`stderr`). Í C++ geturðu notað `cerr << msg << endl;`. Í Python geturðu notað `print(msg, file=sys.stderr)`.

Prófunartólið les og sýnir þessi `stderr` skilaboð ásamt fyrirspurnunum sem öll forritstilvikin þín gera. Athugaðu að af tæknilegum ástæðum gætu þau birst aðeins úr takti við hvort annað.