

Loja në klasë

Një gjimnaz i njohur në Tashkent e feston përvjetorin e tij duke organizuar një lojë midis nxënësve dhe mësuesve. Në klasë janë ulur N nxënës, të numëruar nga 0 deri në $N - 1$. Çdo nxënës mban një fletë letre që përmban një varg numrash të plotë. Fillimisht, çdo fletë është bosh, pra përmban një varg bosh.

Nxënësit e luajnë lojën me M mësues, të numëruar nga 0 deri në $M - 1$, dhe me drejtorin. Loja luhet në M hapa, të numëruar gjithashtu nga 0 deri në $M - 1$. Në hapin j , mësuesi j hyn në klasë dhe ndodhin ngjarjet e mëposhtme:

- Disa nxënës (mundet edhe asnjë) ngrenë dorën. Garantohet se në çdo hap të paktën një nxënës *nuk* e ngre dorën, dhe se çdo nxënës mund ta ngrejë dorën *të shumtën një herë* gjatë gjithë lojës.
- Mësuesi i shikon fletët që nxënësit mbajnë aktualisht dhe mund t'i *modifikojë* fletët e mbajtura nga nxënësit që *nuk* e ngritën dorën në këtë hap. Për secilin nxënës të tillë, mësuesi mund ta zëvendësojë përmbajtjen e fletës së tij me një varg të ri (mundet edhe bosh) numrash të plotë. Secili numër i plotë duhet të jetë midis 0 dhe 63, duke përfshirë edhe kufijtë, dhe vargu mund të ketë të shumtën 63 elemente.
- Pas këtyre modifikimeve, mësuesi j largohet dhe nxënësit i shkëmbejnë fletët e tyre sipas një permutacioni të fshehtë P . Domethënë, çdo nxënës i ($0 \leq i < N$) ia jep fletën e tij nxënësit $P[i]$, ku $P[0], P[1], \dots, P[N - 1]$ janë N numra të plotë të *ndryshëm* midis 0 dhe $N - 1$, duke përfshirë edhe kufijtë. Vlerat e P janë të *fiksuar* gjatë të gjithë hapave të lojës, por mësuesit dhe drejtori nuk i dinë ato.

Pasi të kenë përfunduar të gjithë M hapat dhe të jetë kryer shkëmbimi i fundit sipas P , hyn drejtori. Duke shikuar vetëm fletët e mbajtura nga nxënësit, drejtori duhet të përcaktojë, për çdo nxënës i ($0 \leq i < N$), numrin e hapit j në të cilin nxënësi i e ngriti dorën, ose të konkludojë se nxënësi i nuk e ngriti kurrë dorën.

Mësuesit punojnë në mënyrë të pavarur. Ata nuk mund të komunikojnë me mësuesit e tjerë ose me drejtorin, përveçse përmes vargjeve të numrave të plotë të shkruar në fletët që nxënësit ia kalojnë njëri-tjetrit.

Detyra juaj është të hartoni dhe të implementoni një strategji për mësuesit dhe drejtorin, e cila identifikon saktë nëse dhe kur secili nxënës e ngriti dorën. Pikët tuaja do të varen nga gjatësia maksimale e çdo vargu të shkruar në një fletë - sa më e shkurtër të jetë kjo gjatësi, aq më të larta janë pikët tuaja.

Detajet e implementimit

Ju duhet të implementoni dy procedura, njëren për mësuesit dhe tjetrën për drejtorin.

Procedura që duhet të implementoni për *mësuesit* është:

```
std::vector<std::vector<int>> process_step(  
    int N, int M, int R,  
    std::vector<int> T,  
    std::vector<std::vector<int>> A)
```

- N : numri i nxënësve.
- M : numri i hapave, si dhe numri i mësuesve.
- R : numri i hapit aktual (nga 0 deri në $M - 1$).
- T : një varg (mundet edhe bosh) i përbërë nga indekset e nxënësve që e ngrenë dorën në këtë hap.
- A : një varg me gjatësi N që përshkruan fletët, ku $A[i]$ ($0 \leq i < N$) është vargu i numrave të plotë në fletën e mbajtur nga nxënësi i në fillim të hapit.
- Kjo procedurë thirret saktësisht M herë për çdo lojë, sipas rendit $R = 0, 1, \dots, M - 1$.

Procedura duhet të kthejë një varg B me gjatësi N , që paraqet fletët pas modifikimeve të mësuesit, në të njëjtin format si A .

- Për çdo nxënës i që e ngriti dorën (domethënë, i shfaqet në T), fleta nuk duhet të ndryshohet, pra $B[i] = A[i]$.
- Për çdo i të tillë që $0 \leq i < N$, gjatësia e $B[i]$ nuk duhet ta kalojë 63, dhe secili numër i plotë në $B[i]$ duhet të jetë midis 0 dhe 63, duke përfshirë edhe kufijtë.

Procedura që duhet të implementoni për *drejtorin* është:

```
std::vector<int> determine_steps(  
    int N, int M, std::vector<std::vector<int>> A)
```

- N, M : njësoj si më sipër.
- A : një varg me gjatësi N , ku $A[i]$ është vargu i numrave të plotë në fletën e mbajtur nga nxënësi i pas të gjithë M hapave.
- Kjo procedurë thirret saktësisht një herë për çdo lojë, pas thirrjes së fundit të process_step.

Procedura duhet të kthejë një varg D me gjatësi N . Për çdo i të tillë që $0 \leq i < N$, duhet të vlejë që

- $D[i] = j$ nëse nxënësi i e ngriti dorën gjatë hapit j , ose
- $D[i] = -1$ nëse nxënësi i nuk e ngriti kurrë dorën gjatë gjithë lojës.

Programi juaj nuk lejohet të ruajë ose të transmetojë asnjë informacion nga një thirrje e `process_step` në një tjetër, përveçse përmes vlerës së kthyer të `process_step`. Nëse programi juaj përpiket ta bëjë këtë, pikët tuaja në CMS mund të jenë të pasakta dhe mund të zvogëlohen pas përfundimit të garës. Vini re se vlerësuesi mund të ekzekutojë disa instanca të programit tuaj njëkohësisht. Kjo do të thotë se thirrjet e `process_step` dhe `determine_steps` nga e njëjta lojë mund të ekzekutohen në instanca të ndryshme, dhe thirrjet e `process_step` dhe `determine_steps` nga lojëra të ndryshme mund të ekzekutohen në të njëjtën instancë. Çdo rast testimi përfshin deri në 5 lojëra.

Kufizimet

- $2 \leq N \leq 63$
- $1 \leq M \leq 63$
- Çdo nxënës e ngre dorën të shumtën një herë gjatë gjithë lojës. Me fjalë të tjera, për çdo nxënës i , ka të shumtën një hap j në të cilin i e ngre dorën.
- Në çdo hap, të paktën një nxënës *nuk* e ngre dorën.

Nëndetyrat dhe pikëzimi

Nëndetyra	Pikët	Kufizime shtesë
1	4	$M = 1$
2	6	$N = 2$
3	9	$P[i] = i$ për të gjitha $0 \leq i \leq N - 1$.
4	25	Të shumtën një nxënës e ngre dorën në çdo hap.
5	56	Nuk ka kufizime shtesë.

Për çdo rast testimi, pikët tuaja janë 0 (raportuar si Output isn't correct në CMS) nëse vlera e kthyer e ndonjë thirrjeje të procedurës `process_step` është e pavlefshme ose nëse vlera e kthyer e ndonjë thirrjeje të procedurës `determine_steps` është e pasaktë.

Përndryshe, le të jetë C gjatësia maksimale e *çfarëdo* vargu në *çfarëdo* B të kthyer nga `process_step`. Atëherë, pikët për një rast testimi në një nëndetyrë me pikë S llogariten si $S \cdot X$, ku X varet nga C sipas tabelës së mëposhtme:

Kushti	X
$C \leq 2$	1.00
$C = 3$	0.75
$C = 4$	0.55
$5 \leq C \leq 13$	$0.50 - 0.03 \cdot (C - 5)$
$14 \leq C \leq 63$	$0.19 \cdot \frac{64-C}{64-14} + 0.04$

Shembull

Shqyrtoni një skenar me $N = 4$ nxënës, $M = 2$ mësues dhe permutacion $P = [0, 3, 1, 2]$. Fillimisht, të gjitha fletët janë bosh.

Vlerësuesi fillimisht thërret:

```
process_step(4, 2, 0, [1], [[], [], [], []])
```

Nxënësi 1 e ka ngritur dorën, kështu që fleta e tij nuk mund të modifikohet. Mësuesi 0 mund të vendosë të shkruajë [10] në fletën e nxënësit 0, të shkruajë [1, 63, 4] në fletën e nxënësit 3 dhe ta lërë bosh fletën e nxënësit 2. Për ta bërë këtë, procedura duhet të kthejë $B = [[10], [], [], [1, 63, 4]]$.

Pasi mësuesi 0 largohet, nxënësit i shkëmbejnë fletët e tyre sipas P . Pas shkëmbimit, fletët janë $A = [[10], [], [1, 63, 4], []]$.

Vlerësuesi tani thërret:

```
process_step(4, 2, 1, [0, 3], [[10], [], [1, 63, 4], []])
```

Nxënësit 0 dhe 3 e kanë ngritur dorën, kështu që fletët e tyre nuk mund të modifikohen. Mësuesi 1 mund të vendosë të shkruajë [0, 40] në fletën e nxënësit 1 dhe të shkruajë [1, 50] në fletën e nxënësit 2. Për ta bërë këtë, procedura duhet të kthejë $B = [[10], [0, 40], [1, 50], []]$.

Pasi mësuesi 1 largohet, fletët shkëmbehen sërish sipas P . Pas shkëmbimit, fletët janë $A = [[10], [1, 50], [], [0, 40]]$.

Së fundi, vlerësuesi thërret:

```
determine_steps(4, 2, [[10], [1, 50], [], [0, 40]])
```

Kjo procedurë duhet të kthejë vargun $D = [1, 0, -1, 1]$, meqenëse nxënësit 0 dhe 3 e ngritën dorën në hapin 1, nxënësi 1 e ngriti dorën në hapin 0 dhe nxënësi 2 nuk e ngriti kurrë dorën. Në këtë

shembull, $C = 3$.

Vlerësuesi shembull

Formati i hyrjes:

```
N M
Q[0] Q[1] ... Q[N-1]
P[0] P[1] ... P[N-1]
```

Këtu, Q është një varg me gjatësi N , ku $Q[i] = j$ nëse nxënësi i e ngre dorën gjatë hapit j , ose $Q[i] = -1$ nëse nxënësi i nuk e ngre kurrë dorën gjatë gjithë lojës.

Formati i daljes:

Pas çdo thirrjeje të `process_step`, vlerësuesi shembull nxjerr përmbajtjen e fletëve. Le të jetë K gjatësia e B dhe $L[i]$ gjatësia e $B[i]$ (për çdo $0 \leq i < K$). Vlerësuesi shembull e printon B në formatin e mëposhtëm, të ndjekur nga një rresht bosh:

```
K
L[0] B[0][0] B[0][1] ... B[0][L[0]-1]
L[1] B[1][0] B[1][1] ... B[1][L[1]-1]
:
L[K-1] B[K-1][0] B[K-1][1] ... B[K-1][L[K-1]-1]
```

Vlerësuesi shembull më pas i shkëmben fletët sipas permutacionit P . Nëse $K \neq N$, në vend të kësaj vlerësuesi shembull printon një mesazh gabimi dhe ndërpritet.

Pas thirrjes së `determine_steps`, vlerësuesi shembull nxjerr:

```
C H
D[0] D[1] ... D[H-1]
```

Këtu, H është gjatësia e vargut D të kthyer nga `determine_steps`.