

Ангийн тоглоом

Ташкент дахь алдартай нэг дунд сургууль өөрийн сургууль байгуулагдсаны ойг тэмдэглэхдээ сурагч болон багш нарын хооронд тоглоом тоглуулахаар шийдсэн. Ангид 0-ээс $N - 1$ хүртлэх тоонуудаар дугаарлагдсан N ширхэг сурагч суудаг. Сурагч бүрт цаас байх ба түүн дээр бүхэл тоон массив бичсэн байгаа. Анх сурагч бүрийн цаас хоосон байх тул түүн дээр байгаа массив нь хоосон массив байна.

Сурагчид 0-ээс $M - 1$ хүртлэх тоонуудаар дугаарлагдсан M ширхэг багш нар болон хичээлийн эрхлэгчтэй хамт тоглоомоо тоглоно. Тоглоом нь 0-ээс $M - 1$ хүртлэх тоонуудаар дугаарлагдсан M ширхэг алхмаас тогтоно. j -р алхамд j -р багш ангид орж ирэх ба доорх үйл явдлууд болно:

- Хэд хэдэн (магадгүй тэг ширхэг) сурагч гараа өргөнө. Алхам бүрийн хувьд дор хаяж нэг сурагч гараа **өргөхгүй** бөгөөд сурагч бүр тоглолтын туршид гараа **дээд тал нь нэг удаа** өргөнө гэдэг нь баталгаатай байна.
- Багш сурагчдад байгаа цааснуудыг үзэх ба энэ алхамд гараа **өргөөгүй** сурагчдын цаасан дээр өөрчлөлт оруулж болно. Ийм сурагч бүрийн хувьд багш цаасан дээрх массивыг нь өөр (хоосон ч байж болох) бүхэл тоон массиваар сольж болно. Бүхэл тоо бүр нь 0-ээс 63 (энэ тоо ороод) хүртлэх тоо байх ба массив нь хамгийн ихдээ 63 элементтэй байна.
- Эдгээр өөрчлөлтүүдийн дараа j -р багш ангиас гарч явах ба сурагч нар өөрсдийн цаасаа нууц P сэлгэмлийн дагуу солилцоно. Энэ нь i -р ($0 \leq i < N$) сурагч бүр өөрийн цаасаа $P[i]$ -р сурагчид өгөх ба энд $P[0], P[1], \dots, P[N - 1]$ нь 0-ээс $N - 1$ (энэ тоог оруулаад) хүртлэх завсраас утгаа авах N ширхэг **ялгаатай** бүхэл тоонууд юм. P -гийн утгууд нь тоглолтын бүх алхмын турш **бэхлэгдсэн** байх боловч багш болон хичээлийн эрхлэгч тэдгээрийг мэдэхгүй.

Бүх M алхмууд дуусч, P -гийн дагуу хамгийн солилцоо хийгдсэний дараа ангид хичээлийн эрхлэгч орж ирнэ. Сурагчийн гарт байгаа цаасыг хараад хичээлийн эрхлэгч i -р ($0 \leq i < N$) сурагч бүрийн хувьд уг сурагчийн гараа өргөсөн алхмын дугаар j -г тодорхойлох эсвэл i -р сурагч огт гар өргөөгүй болохыг тодорхойлно.

Багш нар бусад багштайгаа болон хичээлийн эрхлэгчтэй зөвхөн цаасан дээр байгаа массиваар дамжуулахаас өөрөөр харилцаж чадахгүй. Багш бүр аль алхам дээр ангид орсноо мэдэж байгаа.

Таны даалгавар бол багш нар болон сургуулийн захирлын хувьд сурагч бүр аль аль алхамд, хэзээ гараа өргөснийг зөв тодорхойлох стратеги боловсруулж, хэрэгжүүлэх явдал юм. Таны

авах оноо цаасан дээрх массивуудын уртаас хамаарна. Максимум урт бага байх тусам таны оноо өндөр эсвэл хэвээрээ байна.

Хэрэгжүүлэлтийн дэлгэрэнгүй мэдээлэл

Та багш нарт зориулсан болон хичээлийн эрхлэгчид зориулсан гэсэн хоёр процедурыг хэрэгжүүлнэ.

Багш нарт зориулсан таны хэрэгжүүлэх процедур нь доорх хэлбэртэй байна:

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

- N : оюутны тоо.
- M : алхмын тоо буюу мөн багш нарын тоо.
- R : одоогийн алхмын дугаар (0-ээс $M - 1$ -ийн хооронд).
- T : тухайн алхамд гараа өргөсөн оюутнуудын индексийг агуулсан (хоосон байж болох), өсөхөөр эрэмбэлэгдсэн массив.
- A : цааснуудыг илэрхийлэх N урттай массив. Энд $A[i]$ ($0 \leq i < N$) нь уг алхамын эхэнд i дугаар сурагчын цаасан дээр байгаа бүхэл тоон массив
- Энэ процедурыг тоглолт бүр дээр яг M удаа, $R = 0, 1, \dots, M - 1$ дарааллаар дуудна.

Уг процедур нь багшийн өөрчлөлтийн дараах цааснуудыг A -тай адил форматаар илэрхийлэх, N урттай B массивыг буцаана.

- Гараа өргөсөн i дугаартай сурагч бүрийн хувьд (энэ нь i утга T дотор агуулагдана гэсэн үг) цаас өөрчлөгдөхгүй буюу $B[i] = A[i]$ байна.
- $0 \leq i < N$ байх i бүрийн хувьд $B[i]$ -гийн урт 63-аас хэтрэхгүй байх ба $B[i]$ дахь бүхэл тоо бүр 0-ээс 63 (энэ тоог оруулаад) гэсэн завсарт байна.

Хичээлийн эрхлэгчид зориулан таны хэрэгжүүлэх процедур нь доорх хэлбэртэй байна:

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

- N, M : дээр өгүүлсэнтэй адил.
- A : N урттай массив. Энд $A[i]$ нь i -р сурагчид бүх M алхмын дараа байх цаасан дээрх бүхэл тоон массив юм.
- Энэ процедурыг тоглолт бүрийн хувьд яг нэг удаа, `process_step` процедурын эцсийн дуудалтын дараа дуудна.

Уг процедур N урттай D массивыг буцаана. $0 \leq i < N$ байх i бүрийн хувьд дараах нөхцөл биелнэ:

- $D[i] = j$ хэрэв i -р сурагч j -р алхмын үед гараа өргөсөн бол, эсвэл
- $D[i] = -1$ хэрэв i -р сурагч тоглолтын бүхий л хугацаанд хэзээ ч гараа өргөөгүй бол.

Таны программ утга буцаахаас өөр байдлаар `process_step` процедурын дуудалтуудын хооронд ямар ч мэдээллийг хадгалах болон дамжуулахыг хориглоно. Хэрэв таны программ үүнийг хийхийг оролдвол таны оноо CMS дээр буруу гарч, тэмцээний төгсгөлд оноо буурч болно. Шалгагч таны программын олон хуулбарыг зэрэг ажиллуулж болохыг анхаараарай. Энэ нь нэг тоглолтын `process_step` болон `determine_steps` дуудалтуудыг өөр өөр хуулбарт ажиллуулж болох ба ялгаатай тоглолтын `process_step` ба `determine_steps` дуудалтуудыг нэг хуулбарт ажиллуулж болно. Тест бүр 5 хүртлэх тоглолтыг агуулж болно.

Хязгаарлалт

- $2 \leq N \leq 63$
- $1 \leq M \leq 63$
- Тоглолтын турш сурагч бүр гараа **дээд тал нь нэг удаа** өргөнө. Өөрөөр хэлбэл, i -р сурагч бүрийн хувьд түүний гараа өргөсөн j -р алхам дээд тал нь нэг оршин байна.
- Алхам бүр дээр дор хаяж нэг сурагч гараа **өргөхгүй** байна.

Дэд бодлого ба Оноо

Дэд бодлого	Оноо	Нэмэлт хязгаарлалт
1	4	$M = 1$
2	6	$N = 2$
3	9	$P[i] = i$ (бүх $0 \leq i \leq N - 1$ утгын хувьд).
4	25	Алхам бүр дээр дээд тал нь нэг сурагч гараа өргөнө.
5	56	Нэмэлт хязгаарлалт байхгүй.

Тест бүрийн хувьд хэрэв `process_step` процедурын дуудалт бүрийн буцаах утга буруу эсвэл `determine_steps` процедурын дуудалт бүрийн буцаах утга буруу бол таны оноо 0 (CMS дээр `Output isn't correct` гэсэн мэдээлэл гарна) байна.

Эсрэг тохиолдолд `process_step` процедурын буцаасан **дурын** B массив дахь **дурын** массивын уртын максимум утгыг C гэж тэмдэглэе. Тэгвэл S оноотой дэд бодлогоос авах оноог $S \cdot X$ гэж тооцоолно. Энд X -г C -гээс доорх хүснэгтийг ашиглан гаргана :

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

Жишээ

$N = 4$ сурагч, $M = 2$ багш, $P = [0, 3, 1, 2]$ сэлгэмэл бүхий тохиолдлыг авч үзье. Анх бүх цаас хоосон байна.

Шалгагчийн эхний дуудалт:

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

1-р сурагч гараа өргөсөн тул түүний цаасыг өөрчлөх боломжгүй.

0-р багш 0-р сурагчийн цаасан дээр [10] гэж, 3-р сурагчийн цаасан дээр [1, 63, 4] гэж бичээд 2-р сурагчийн цаасыг хоосон орхихыг хүсч болно. Ингэхийн тулд уг процедур $B = [[10], [], [], [1, 63, 4]]$ массивыг буцаана.

0-р багшийг явсны дараа сурагч нар P сэлгэмлийн дагуу цаасаа солилцоно. Солилцооны дараа цааснууд $A = [[10], [], [1, 63, 4], []]$ хэлбэртэй байна.

Шалгагчийн дараагийн дуудалт:

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

0 болон 3-р сурагч нар гараа өргөсөн тул тэдгээрийн цаасыг өөрчлөх боломжгүй. 1-р багш 1-р сурагчийн цаасан дээр [0, 40] гэж, 2-р сурагчийн цаасан дээр [1, 50] гэж бичихийг хүсч болно. Ингэхийн тулд уг процедур $B = [[10], [0, 40], [1, 50], []]$ массивыг буцаана.

1-р багш явсны дараа цааснуудыг P -гийн дагуу дахин солилцоно. Солилцооны дараа цааснууд $A = [[10], [1, 50], [], [0, 40]]$ хэлбэртэй болно.

Шалгагчийн сүүлийн дуудалт:

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

0 ба 3-р сурагч нар 1-р алхамд, 1-р сурагч 0-р алхамд гараа өргөсөн ба 2-р сурагч огт гараа өргөөгүй тул уг процедур $D = [1, 0, -1, 1]$ массивыг буцаана. Энэ жишээн дээр $C = 3$ байна.

Жишээ шалгагч

Оролтын формат:

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

Энд Q нь N урттай массив байна. Хэрэв i -р сурагч гараа j -р алхмын үед өргөсөн бол $Q[i] = j$, эсвэл i -р сурагч гараа тоглолтын турш огт өргөөгүй бол $Q[i] = -1$ байна.

Гаралтын формат:

`process_step` процедурын дуудалт бүрийн дараа шалгагч цааснууд дээр байгаа тоонуудыг хэвлэнэ. K нь B -гийн урт ба $L[i]$ нь $B[i]$ -гийн ($0 \leq i < K$) урт байг. Жишээ шалгагч B -г доорх форматаар, **хоосон мөрөөр тусгаарлан** хэвлэнэ:

```
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]
```

Дараа нь жишээ шалгагч цааснуудыг P сэлгэмлийн дагуу солино. Хэрэв $K \neq N$ бол жишээ шалгагч алдаа хэвлээд ажиллагааг зогсооно.

`determine_steps` функцийн дуудалтын дараа шалгагч доорхийг гаргана:

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

Энд, H нь `determine_steps` процедурын буцаасан D массивын урт юм.