

Сыныптағы ойын

Ташкенттегі беделді мектеп өзінің мерейтойын оқушылар мен мұғалімдер арасындағы ойын өткізу арқылы атап өтеді. Сыныпта N оқушы отыр. Олар 0-ден $N - 1$ -ге дейін нөмірленген. Әр оқушының қолында бүтін сандар массиві жазылған қағаз парағы бар. Бастапқыда әр қағазда ештеңе жазылмаған, яғни сол қағаздағы массив бос.

Оқушылар ойынды 0-ден $M - 1$ -ге дейін нөмірленген M мұғалімдермен және мектеп директорымен бірге ойнайды. Ойын M қадамда өтеді. Олар да 0-ден $M - 1$ -ге дейін нөмірленген. j -ші қадамда j -ші мұғалім сыныпқа кіреді және келесі оқиғалар орын алады:

- Кейбір (мүмкін ешбір) оқушылар қол көтереді. Әрбір қадамда кем дегенде бір оқушы қол көтермейтіні кепілдендірілген, ал әрбір оқушы бүкіл ойын барысында тек бір рет қана қол көтере алады.
- Мұғалім оқушылардың қазір ұстап отырған қағаздарына қарап, осы қадамда қол көтермеген оқушылардың қағаздарын **өзгерте** алады. Әрбір осындай оқушы үшін мұғалім олардың қағазындағы мазмұнды жаңа (мүмкін бос) бүтін сандар массивімен алмастыра алады. Әрбір бүтін сан 0 мен 63 аралығында (қоса алғанда) болуы тиіс және массивте ең көбі 63 элемент болуы мүмкін.
- Осы өзгерістерден кейін j -ші мұғалім шығып кетеді және оқушылар қағаздарын құпия P алмастыруы бойынша айырбастайды. Яғни, әрбір i -ші оқушы ($0 \leq i < N$) өз қағазын $P[i]$ -ші оқушыға береді, мұнда $P[0], P[1], \dots, P[N - 1]$ алмастыруы – 0 мен $N - 1$ аралығындағы (қоса алғанда) бір-біріне ұқсамайтын N бүтін сандар. P -нің мәндері ойынның барлық кезеңдерінде **бекітілген**, бірақ мұғалімдер мен директор оларды білмейді.

Барлық M қадамдар аяқталғаннан және соңғы P алмасуы жүзеге асырылғаннан кейін директор кіреді. Тек оқушылардың ұстап тұрған қағаздарына қарап, директор әрбір i -ші оқушы үшін ($0 \leq i < N$) i -ші оқушы қол көтерген j қадам нөмірін анықтауы тиіс, немесе i -ші оқушы ешқашан қол көтермегенін анықтау керек.

Мұғалімдер басқа мұғалімдермен немесе мектеп директорымен қағаздарға жазылған бүтін сандар массивтері арқылы ғана байланыса алады. Әрбір мұғалім сыныпқа кіретін қадамын біледі.

Сіздің міндетіңіз – әрбір оқушының қашан қол көтергенін дұрыс анықтайтын мұғалімдер мен директорға арналған стратегияны ойлап тауып, іске асыру. Сіздің ұпайыңыз қағазға жазылған кез келген массивтің максималды ұзындығына байланысты болады – қысқарақ максималды ұзындық жоғары немесе тең ұпай алуға әкеледі.

Implementation Details

Сізге екі функцияны іске асыру керек: біреуі мұғалімдерге, екіншісі директорға.

Мұғалімдер үшін іске асыруыңыз керек функция мынадай:

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

- N : оқушылар саны.
- M : қадамдар және мұғалімдер саны.
- R : ағымдағы қадам нөмірі (0-ден $M - 1$ -ке дейін).
- T : осы қадамда қолын көтерген оқушылардың индекстерінен тұратын өсу ретіндегі массив (мүмкін бос).
- A : ұзындығы N болатын қағаздар массиві, мұнда $A[i]$ ($0 \leq i < N$) – қадам басталған кезде i -ші оқушының қолындағы қағаздағы бүтін сандар массиві.
- Бұл функция әр ойын сайын дәл M рет $R = 0, 1, \dots, M - 1$ ретімен шақырылады.

Функция ұзындығы N болатын B массивін қайтаруы тиіс, ол мұғалімнің түзетулерінен кейінгі қағаздарды сипаттайды. Формат A -мен бірдей.

- Қол көтерген әрбір i -ші оқушы үшін (яғни i T тізімінде бар), жұмыс өзгертілмеуі тиіс, сондықтан $B[i] = A[i]$.
- $0 \leq i < N$ шарты орындалатын әрбір i үшін $B[i]$ ұзындығы 63-ден аспауы тиіс, ал $B[i]$ -дегі барлық бүтін сандар 0-тен 63-ке дейінгі аралықта (қоса алғанда) болуы тиіс.

Директор үшін іске асыруыңыз керек функция мынадай:

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

- N, M : жоғарыдағыдай.
- A : ұзындығы N болатын массив, мұнда $A[i]$ – барлық M қадамнан кейін i -ші оқушының қағазындағы бүтін сандар массиві.
- Бұл функция әр ойын сайын `process_step`-ке соңғы шақырудан кейін дәл бір рет шақырылады.

Функция N ұзындықтағы D массивін қайтаруы тиіс. $0 \leq i < N$ болатын әрбір i үшін келесі шарттар орындалуы тиіс:

- $D[i] = j$, егер i -ші оқушы j кезеңінде қолын көтерген болса, немесе
- $D[i] = -1$, егер i -ші оқушы бүкіл ойын бойы қолын ешқашан көтермеген болса.

Сіздің бағдарламаңызға `process_step`-тің қайтару мәнінен басқа `process_step`-ке жасалған бір шақырудан екінші шақыруға ешқандай ақпаратты сақтауға немесе беруге рұқсат етілмейді. Егер бағдарламаңыз мұны істеуге тырысса, CMS-тегі ұпайыңыз дұрыс есептелмеуі мүмкін және жарыс аяқталғаннан кейін азайтылуы мүмкін. Ескерту: грейдер сіздің бағдарламаңызды бірнеше рет шақыра алады. Бұл бір ойыннан `process_step` және `determine_steps` шақырулары әртүрлі бағдарламада қосылуында орындалуы мүмкін екенін, ал әртүрлі ойындардан `process_step` және `determine_steps` шақырулары бірдей бағдарламада шақыруында орындалуы мүмкін екенін білдіреді. Әр тест жағдайы 5 ойынға дейін қамтиды.

Constraints

- $2 \leq N \leq 63$
- $1 \leq M \leq 63$
- Ойын барысында әр оқушы ең көбі бір рет қолын көтереді. Басқаша айтқанда, әрбір i үшін i -ші оқушы қолын көтеретін ең көбі бір ғана j қадамы бар.
- Әрбір қадамда кемінде бір оқушы қолын көтермейді.

Subtasks and Scoring

Ішкі есеп	Ұпай	Қосымша шектеулер
1	4	$M = 1$
2	6	$N = 2$
3	9	$P[i] = i$ for all $0 \leq i \leq N - 1$.
4	25	Әр қадамда ең көбі бір оқушы қолын көтереді.
5	56	Қосымша шектеулер жоқ.

Әрбір тест үшін егер `process_step` функциясына жасалған кез келген шақырудың қайтарылған мәні жарамсыз болса немесе `determine_steps` функциясына жасалған кез келген шақырудың қайтарылған мәні дұрыс болмаса, сіздің ұпайыңыз 0 болады (CMS жүйесінде `Output isn't correct` деп көрсетіледі).

Әйтпесе, `process_step` функциясы қайтаратын кез келген B массивінің ең үлкен ұзындығын C деп белгілейік. Содан кейін S ұпайы бар ішкі есептегі тест ұпайы $S \cdot X$ формуласы бойынша есептеледі, мұндағы X мәні C мәніне келесі кестеге сәйкес тәуелді:

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

Example

$N = 4$ оқушылары, $M = 2$ мұғалімдері және алмастыру $P = [0, 3, 1, 2]$ бар жағдайды қарастырыңыз. Бастапқыда барлық қағаздар бос.

Грейдер алдымен келесіні шақырады:

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

1-ші оқушы қолын көтерді, сондықтан оның қағазын өзгертуге болмайды. 0-ші мұғалім 0-ші оқушының қағазына $[10]$ деп жазуды, 3-ші оқушының қағазына $[1, 63, 4]$ деп жазуды және 2-ші оқушының қағазын бос қалдыруын шешуі мүмкін. Мұны іске асыру үшін функция $B = [[10], [], [], [1, 63, 4]]$ мәнін қайтаруы тиіс.

0-ші мұғалім кеткеннен кейін, оқушылар P бойынша қағаздарын алмастырады. Алмастырудан кейін, қағаздар $A = [[10], [], [1, 63, 4], []]$ болады.

Грейдер енді келесіні шақырады:

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

0-ші және 3-ші оқушылар өз қолдарын көтергендіктен, олардың қағаздарын өзгертуге болмайды. 1-ші мұғалім 1-ші оқушының жұмысына $[0, 40]$, ал 2-ші оқушының жұмысына $[1, 50]$ жазуды шешуі мүмкін. Мұны іске асыру үшін функция $B = [[10], [0, 40], [1, 50], []]$ мәнін қайтаруы тиіс.

1-ші мұғалім кеткеннен кейін жұмыстар P -ға сәйкес қайта алмастырылады. Алмасудан кейін қағаздарда $A = [[10], [1, 50], [], [0, 40]]$ болады.

Соңында грейдер келесіні шақырады:

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

Бұл функция $D = [1, 0, -1, 1]$ массивін қайтаруы тиіс, себебі 0-ші және 3-ші оқушылар 1-ші қадамда қолын көтерді, 1-ші оқушы 0-ші қадамда қолын көтерді, ал 2-ші оқушы ешқашан қолын көтермеді. Бұл мысалда $C = 3$.

Sample Grader

Input format:

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

Мұнда Q – ұзындығы N болатын массив, мұнда $Q[i] = j$ – егер i -ші оқушы j -ші қадам кезінде қолын көтерсе, немесе $Q[i] = -1$ – егер i -ші оқушы бүкіл ойын бойы қолын көтермесе.

Output format:

`process_step`-ке әрбір шақырудан кейін үлгі грейдер қағаздардың мазмұнын шығарады. K мәнін B массивінің ұзындығы деп, ал $L[i]$ мәнін $B[i]$ массивінің ұзындығы деп белгілейік (әрбір $0 \leq i < K$ үшін). Үлгі грейдер B массивін келесі форматта басып шығарады, **одан кейін бос жол болады**:

```
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]
```

Үлгі грейдер содан кейін қағаздарды P алмастыруына сәйкес алмастырады. Егер $K \neq N$ болса, үлгі грейдер қате хабарламасын басып шығарып, жұмысты тоқтатады.

`determine_steps`-ты шақырғаннан кейін, үлгі грейдер келесі шығаруды береді:

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

Мұнда H – `determine_steps` функциясының қайтарған D массивінің ұзындығы.