

Classroom Game

Toshkentdagi taniqli o'rta maktab o'zining yubileyini o'quvchilar va o'qituvchilar o'rtasida o'yin o'tkazish orqali nishonlamoqda. Sinf xonasida 0 dan $N - 1$ gacha raqamlangan N ta talaba o'tirishibdi. Har bir talaba butun sonlar massivini o'z ichiga olgan qog'oz varag'ini ushlab turibdi. Dastlab, har bir qog'oz bo'sh, shuning uchun u bo'sh massivdan iborat bo'ladi.

O'quvchilar o'yinni 0 dan $M - 1$ gacha raqamlangan M ta o'qituvchi va direktor bilan o'ynashmoqda. O'yin M ta bosqichda o'ynaladi, ular ham 0 dan $M - 1$ gacha raqamlangan. j -bosqichda, j -o'qituvchi sinfga kiradi va quyidagilar sodir bo'ladi:

- Ba'zi (yoki 0 ta) o'quvchilar qo'llarini ko'tarishadi. Har bir qadamda kamida bitta o'quvchi qo'lini **ko'tarmasligi** kafolatlanadi va har bir o'quvchi butun o'yin davomida qo'lini **ko'pi bilan bir marta** ko'tarishi mumkin.
- O'qituvchi talabalarning hozirda ushlab turgan qog'ozlarini ko'rib chiqadi va hozirgi bosqichda qo'lini **ko'tarmagan** talabalarning qog'ozlarini **o'zgartirishi** mumkin. Har bir talaba uchun o'qituvchi qog'oz tarkibini yangi (bo'sh bo'lishi ham mumkin) butun sonlar massivi bilan almashtirishi mumkin. Har bir butun son 0 va 63 orasida (inklusiv) bo'lishi kerak va massivda eng ko'pi bilan 63 ta element bo'lishi mumkin.
- Ushbu o'zgartirishlardan so'ng, j -o'qituvchi ketadi va talabalar yashirin permutatsiya P ga muvofiq qog'ozlarini almashadilar. Ya'ni, har bir i -talaba ($0 \leq i < N$) o'z ishini $P[i]$ - talabaga beradi, bu yerda $P[0], P[1], \dots, P[N - 1]$ - bu N ta 0 va $N - 1$ oralig'idagi **har xil** butun sonlardir. P ning qiymatlari o'yinning barcha bosqichlarida **o'zgarmas** bo'ladi, lekin o'qituvchilar va direktor P ni bilishmaydi.

Barcha M ta bosqich tugallangandan so'ng va P bo'yicha oxirgi almashtirish amalga oshirilgandan so'ng, direktor ichkariga kiradi. Faqat talabalar qo'lida ushlab turgan qog'ozlarga qarab, direktor har bir i -talaba ($0 \leq i < N$) uchun i -talaba qo'lini ko'targan bosqich raqami j ni aniqlashi yoki i talaba hech qachon qo'lini ko'tarmaganini topishi kerak.

O'qituvchilar boshqa o'qituvchilar yoki maktab direktori bilan muloqot qila olmaydilar (qog'ozlarga yozilgan massivlardan tashqari). Har bir o'qituvchi qaysi bosqichda sinfga kirganini bilishadi.

Sizning vazifangiz o'qituvchilar va direktor uchun har bir o'quvchi qo'lini ko'targanmi va ko'targan bo'lsa, qaysi bosqichda ko'targanini to'g'ri aniqlaydigan strategiyani ishlab chiqish va amalga oshirishdir. Sizning ballingiz qog'ozga yozilgan har qanday massivning maksimal uzunligiga bog'liq bo'ladi: qisqaroq maksimal uzunlik yuqori yoki teng ball beradi.

Implementation Details

Siz ikkita protsedurani shakllantirishingiz kerak, biri o'qituvchilar uchun, ikkinchisi direktor uchun.

O'qituvchilar uchun yozib chiqishingiz kerak bo'lgan protsedura quyidagicha:

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

- N : talabalar soni.
- M : qadamlar soni va o'qituvchilar soni.
- R : joriy qadam raqami (0 dan $M - 1$ gacha).
- T : joriy bosqichda qo'l ko'targan talabalar indekslaridan iborat (bo'sh bo'lishi ham mumkin) massiv.
- T : ushbu bosqichda qo'l ko'targan talabalarining indekslaridan iborat (ehtimol bo'sh) massiv bo'lib, ular o'suvchi tartibda saralanadi.
- Bu protsedura har bir o'yin uchun aniq M marta, $R = 0, 1, \dots, M - 1$ tartibida chaqiriladi.

Protsedura o'qituvchining o'zgartirishlaridan keyingi qog'ozlarni ifodalovchi N uzunlikdagi B massivini A bilan bir xil formatda qaytarishi kerak.

- Qo'lini ko'targan har bir i -talaba uchun (ya'ni, T ning ichidagi i), qog'oz o'zgartirilmaligi kerak, shuning uchun $B[i] = A[i]$.
- Har bir i ($0 \leq i < N$) uchun, $B[i]$ ning uzunligi 63 dan oshmasligi kerak va $B[i]$ dagi har bir butun son 0 va 63 orasida (inklusiv) bo'lishi kerak.

Direktor uchun yozib chiqishingiz kerak bo'lgan protsedura:

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

- N, M : yuqoridagi bilan bir xil.
- A : N uzunlikdagi massiv, bu yerda $A[i]$ - i -talaba M ta bosqichdan keyin ushlab turgan butun sonlar massivi.
- Ushbu protsedura har bir o'yinda bir marta, `process_step` ga oxirgi chaqiruvdan so'ng chaqiriladi.

Protsedura N uzunlikdagi D massivini qaytarishi kerak. Har bir i ($0 \leq i < N$) uchun u quyidagilarni saqlashi kerak

- Agar i -talaba j -bosqichda qo'lini ko'targan bo'lsa $D[i] = j$, yoki
- Agar i -talaba butun o'yin davomida qo'lini ko'tarmagan bo'lsa, $D[i] = -1$.

Dasturingiz bir chaqiruvdan `process_step` ga boshqasiga hech qanday ma'lumotni saqlashi yoki uzatishi mumkin emas, muloqot faqat `process_step` qaytaradigan qiymat orqali amalga oshiriladi. Agar dasturingiz buni qilishga harakat qilsa, CMSdagi kontest paytida chiqqan ballingiz noto'g'ri bo'lishi mumkin va kontest tugaganidan keyin kamaytirilishi mumkin. E'tibor bering, grader dasturingizni bir vaqtning o'zida bir nechta protsesslarda bajarishi mumkin. Bu shuni anglatadiki, bir xil o'yindan `process_step` va `determine_steps` ga chaqiruvlar turli protsesslarda bajarilishi mumkin va turli o'yinlardan `process_step` va `determine_steps` ga chaqiruvlar bir xil protsesslarda bajarilishi mumkin. Har bir test 5 tagacha bo'lgan o'yinni o'z ichiga oladi.

Constraints

- $2 \leq N \leq 63$
- $1 \leq M \leq 63$
- Har bir talaba butun o'yin davomida **ko'pi bilan bir marta** qo'lini ko'taradi. Boshqacha qilib aytganda, har bir i -talaba uchun qo'lini ko'taradigan ko'pi bilan bitta j -qadam mavjud.
- Har bir qadamda kamida bitta talaba qo'lini ko'tarmaydi.

Subtasks and Scoring

Subtask	Score	Qo'shimcha Cheklovlar
1	4	$M = 1$
2	6	$N = 2$
3	9	Har bir i ($0 \leq i \leq N - 1$) uchun $P[i] = i$.
4	25	Har bir bosqichda ko'pida bitta o'quvchi qo'lini ko'tarishi mumkin.
5	56	Qo'shimcha cheklovlarsiz

Har bir test uchun, sizning ballingiz 0 ni tashkil qiladi, agar `process_step` protsedurasiga biron bir chaqiruvning qaytish qiymati noto'g'ri (shartlarni buzadigan) bo'lsa yoki `determine_steps` protsedurasiga biron bir chaqiruvning qaytish qiymati noto'g'ri bo'lsa, (CMSda `Output isn't correct` deb ko'rsatiladi).

Aks holda, $C_{process_step}$ qaytargan **har qanday** B dagi **har bir** massivning maksimal uzunligi bo'lsin. Keyin, S balliga ega bo'lgan subtaskdagi test uchun ball $S \cdot X$ bilan hisoblanadi, bu yerda X quyidagi jadvalga muvofiq C ga bog'liq son:

Holat	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

$N = 4$ ta talaba, $M = 2$ ta o'qituvchi va $P = [0, 3, 1, 2]$ permutatsiyasi bo'lgan holatni ko'rib chiqaylik. Dastlab, barcha qog'ozlar bo'sh.

Grader boshida quyidagini chaqiradi:

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

1-talaba qo'lini ko'tardi, shuning uchun uning qog'ozini o'zgartirib bo'lmaydi. O'qituvchi 0-talabaning qog'oziga $[10]$ yozishga, 3-talabaning qog'oziga $[1, 63, 4]$ yozishga va 2-talabaning qog'ozini bo'sh qoldirishga qaror qilishi mumkin. Buning uchun protsedura $\$B = [[10], [], [], [1, 63, 4]]\$$ qaytarishi kerak.

0-o'qituvchi ketganidan so'ng, talabalar P ga muvofiq qog'ozlarini almashadilar. Almashishdan so'ng, qog'ozlar $\$A = [[10], [], [1, 63, 4], []]\$$ ga o'zgaradi.

Grader quyidagini chaqiradi:

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

0 va 3-talabalar qo'llarini ko'tarishdi, shuning uchun ularning qog'ozlarini o'zgartirib bo'lmaydi. O'qituvchi 1-talabaning qog'oziga $[0, 40]$ yozishga va 2-talabaning qog'oziga $[1, 50]$ yozishga qaror qilishi mumkin. Buning uchun protsedura $\$B = [[10], [0, 40], [1, 50], []]\$$ qaytarishi kerak.

1-o'qituvchi ketganidan so'ng, qog'ozlar yana P ga muvofiq almashtiriladi. Almashishidan so'ng, qog'ozlar $\$A = [[10], [1, 50], [], [0, 40]]\$$ ga o'zgaradi.

Nihoyat, grader quyidagini chaqiradi:

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

Ushbu protsedura $D = [1, 0, -1, 1]$ massivini qaytarishi kerak, chunki 0 va 3-talabalar 1-bosqichida qo'llarini ko'tarishdi, 1-talaba 0-bosqichda qo'lini ko'tardi va 2-talaba hech qachon qo'lini ko'tarmadi. Ushbu misolda, $C = 3$.

Sample Grader

Input format:

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

Bu yerda Q - bu N uzunlikdagi massiv, agar i talaba j -bosqichda qo'lini ko'targan bo'lsa, $Q[i] = j$, yoki agar i -talaba butun o'yin davomida hech qachon qo'lini ko'tarmasa, $Q[i] = -1$.

Output format:

`process_step` ga har bir chaqiruvdan so'ng, grader qog'ozlardagi sonlarni chiqaradi. K - bu B ning uzunligi va $L[i]$ - bu $B[i]$ massivning uzunligi bo'lsin, har bir i ($0 \leq i < K$) uchun. Namunaviy baholovchi B ni quyidagi formatda, **oxirida bo'sh qator** bilan chop etadi:

```
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]
```

Keyin grader hujjatlarni P permutatsiyasiga muvofiq almashtiradi. Agar $K \neq N$ bo'lsa, grader xatolik haqida habar beradi va ishlashdan to'xtaydi.

`determine_steps` chaqirilgandan so'ng, grader quyidagini chiqaradi:

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

Bu yerda H - bu `determine_steps` tomonidan qaytarilgan D massivining uzunligi.