

Kitob javoni (bookshelf)

Karimjon Dadorlandtiria Milliy Kutubxonasida ishlaydi. Karimjon ishlaydigan bo'limda o'lchami $n \times m$ bo'lgan ulkan kitob javoni bor, bu yerda qatorlar 0 dan $n - 1$ gacha, ustunlar 0 dan $m - 1$ gacha raqamlangan.

Javonning (i, j) katagida $a[i][j]$ raqamli kitob joylashgan. Javonda barcha kitoblarning raqamlari har xil.

Karimjon quyidagi ikki amallarni bajarishi mumkin:

- Barcha **qator** uchun, shu qatordagi kitoblarni ixtiyoriy tartibda joylashtirish. Har bir qator uchun tartibni mustaqil ravishda tanlash mumkin.
- Barcha **ustun** uchun, shu ustundagi kitoblarni ixtiyoriy tartibda joylashtirish. Har bir ustun uchun tartibni mustaqil ravishda tanlash mumkin.

Karimjon javondagi kitoblarni tartiblamoqchi. Tartiblangan javonda 0-qatorda raqamlari 0 dan $m - 1$ gacha bo'lgan barcha kitoblar o'sish tartibida, 1-qatorda raqamlari m dan $2m - 1$ gacha bo'lgan barcha kitoblar o'sish tartibida va h.k. joylashgan bo'llishi kerak. Ya'ni, barcha $0 \leq i \leq n - 1$ va $0 \leq j \leq m - 1$ uchun, $a[i][j] = i \cdot m + j$ shart bajarilishi kerak.

Aytaylik, R bu javonni tartiblash uchun minimal amallar soni bo'lsin. Shuningdek, sizga T soni beriladi, $1 \leq T \leq 2$.

- Agar $T = 1$ bo'lsa, bajarish kerak bo'lgan minimal amallar soni, ya'ni R ni topishingiz kerak.
- Agar $T = 2$ bo'lsa, R ning qiymatini topishingiz hamda bajariladigan amallarni ko'rsatishingiz kerak.

Implementation details

Siz quyidagi protsedurani dasturlashingiz kerak:

```
int sort_books(int n, int m, int T, vector<vector<int>> a)
```

- n : jami qatorlar soni.
- m : jami ustunlar soni.
- a : o'lchamlari $n \times m$ bo'lgan matritsa – kitob javoni haqida ma'lumot.
- Bu protseda butun son – javonni tartiblash uchun minimal amallar soni, ya'ni R ni qaytarishi kerak.

- Bu protsedura aynan bir marta chaqiriladi.

Bu protsedura ichida siz quyidagi protseduralarga murojaat qila olasiz:

```
void rearrange_row(vector<vector<int>> b)
```

- b : o'lchamlari $n \times m$ bo'lgan matritsa – **qator** bo'yicha tartiblash amalini bajargandan so'ng kitob javonining ko'rinishi.
- Bu yerda, barcha $0 \leq i \leq n - 1$ uchun, $a[i][0], a[i][1], \dots, a[i][m - 1]$ hamda $b[i][0], b[i][1], \dots, b[i][m - 1]$ sonlar to'plamlari bir xil bo'lishi kerak.
- Bu protsedurani bir-necha marta chaqirilishingiz mumkin.

```
void rearrange_column(vector<vector<int>> b)
```

- b : o'lchamlari $n \times m$ bo'lgan matritsa – **ustun** bo'yicha tartiblash amalini bajargandan so'ng kitob javonining ko'rinishi.
- Bu yerda, barcha $0 \leq j \leq m - 1$ uchun, $a[0][j], a[1][j], \dots, a[n - 1][j]$ hamda $b[0][j], b[1][j], \dots, b[n - 1][j]$ sonlar to'plamlari bir xil bo'lishi kerak.
- Bu protsedurani bir-necha marta chaqirilishingiz mumkin.

Examples

Example 1

Quyidagi chaqiruvni ko'raylik:

```
sort_books(2, 2, 2,
           [[1, 0],
            [3, 2]])
```

Bu yerda $T = 2$, demak amallarni ko'rsatish kerak. Ko'rsatish mumkinki, minimal amallar soni $R = 1$. Siz quyidagi amalni bajarishingiz mumkin:

```
rearrange_row([[0, 1],
               [2, 3]])
```

Ya'ni, 0-qator uchun, $[1, 0]$ tartibdagi kitoblarni $[0, 1]$ kabi, 1-qator uchun $[2, 3]$ tartibdagi kitoblarni $[3, 2]$ kabi joylashtirish.

Ushbu chaqiruvdan so'ng `sort_books()` protsedurasi R , ya'ni 1 ni qaytarishi lozim.

Example 2

Quyidagi chaqiruvni ko'raylik:

```
sort_books(2, 3, 1,
           [[0, 4, 2],
            [5, 1, 3]])
```

Bu yerda $T = 1$, demak amallarni ko'rsatish shart emas. Isbotlash mumkinki, minimal amallar soni $R = 2$.

`sort_books()` protsedurasi darhol 2 qaytarishi lozim.

Example 3

Quyidagi chaqiruvni ko'raylik:

```
sort_books(5, 3, 2,
           [[10, 9, 11],
            [14, 5, 12],
            [1, 4, 7],
            [3, 8, 13],
            [2, 6, 0]])
```

Isbotlash mumkinki, minimal amallar soni $R = 3$.

```
rearrange_row(
    [[11, 9, 10],
     [12, 14, 5],
     [4, 7, 1],
     [8, 3, 13],
     [0, 2, 6]])
```

```
rearrange_column(
    [[0, 2, 1],
     [4, 3, 5],
     [8, 7, 6],
     [11, 9, 10],
     [12, 14, 13]])
```

```
rearrange_row(
    [[0, 1, 2],
     [3, 4, 5],
     [6, 7, 8],
     [9, 10, 11],
     [12, 13, 14]])
```

Shundan so'ng `sort_books()` funksiyasi 3 qaytarishi kerak.

Constraints

- $1 \leq n, m \leq 100$
- $1 \leq T \leq 2$
- $0 \leq a[i][j] \leq n \cdot m - 1$
- a matritsaning barcha elementlari har xil

Subtasks

1. (15 ball) $n, m \leq 3$
2. (15 ball) $T = 1$
3. (10 ball) testlarda $R \leq 2$ bo'lishi kafolatlanadi
4. (15 ball) $n \leq 3, m \leq 15$
5. (18 ball) $n, m \leq 15$
6. (27 ball) Qo'shimcha cheklovlar siz.

Sample Grader

Namunaviy grader ma'lumotlarni quyidagi tartibda o'qiydi:

- qator 1: $n \ m \ T$
- qator $2 + i$ ($0 \leq i < n$): $a[i][0] \ a[i][1] \ \dots \ a[i][m - 1]$

Agar $T = 1$ bo'lsa, namunaviy grader javobni quyidagi tartibda chiqaradi:

- qator 1: `sort_books()` protsedurasi qaytargan javob.

Agar $T = 2$ bo'lsa, har bir chaqirilgan `rearrange_row(b)` va `rearrange_column(b)` protseduralari uchun, namunaviy grader javobni quyidagi tartibda chiqaradi:

- qator $1 + i$ ($0 \leq i < n$): $b[i][0] \ b[i][1] \ \dots \ b[i][m - 1]$
- So'nggi qatorda, `sort_books()` protsedurasi qaytargan javob.