



## Carnival Tickets (tickets)

Ringo đang tham dự một lễ hội ở Singapore. Cậu có một số vé dự thưởng trong túi và dự định sẽ sử dụng chúng để chơi tại quầy trò chơi có thưởng. Mỗi vé thuộc một trong  $n$  màu và có in một số nguyên không âm. Các số nguyên in trên các vé khác nhau có thể có cùng giá trị. Theo một luật lệ kỳ quặc của lễ hội,  $n$  đảm bảo luôn là **số chẵn**.

Ringo có  $m$  vé mỗi màu, tổng cộng, cậu có tất cả  $n \cdot m$  vé. Vé thứ  $j$  của màu  $i$  có in giá trị nguyên  $x[i][j]$  ( $0 \leq i \leq n - 1$  và  $0 \leq j \leq m - 1$ ).

Trò chơi được chơi trong  $k$  vòng, các vòng được đánh số từ 0 đến  $k - 1$ . Mỗi vòng được chơi theo thứ tự như sau:

- Từ trong túi mình, Ringo chọn một **tập**  $n$  vé, mỗi vé một màu. Sau đó, cậu đưa tập vé cho người quản trò.
- Người quản trò ghi lại các số nguyên  $a[0], a[1] \dots a[n - 1]$  là giá trị được in trên các vé. Thứ tự của  $n$  số nguyên này là không quan trọng.
- Người quản trò lấy một thẻ đặc biệt từ chiếc hộp may mắn và ghi lại số nguyên  $b$  được in trên thẻ đặc biệt.
- Người quản trò tính các chênh lệch tuyệt đối giữa  $a[i]$  và  $b$  với mỗi  $i$  trong khoảng từ 0 đến  $n - 1$ . Gọi  $S$  là tổng của các chênh lệch tuyệt đối này.
- Với vòng chơi này, người quản trò đưa cho Ringo phần thưởng có giá trị bằng  $S$ .
- Các vé trong tập này sẽ được bỏ đi và không thể được sử dụng cho các vòng chơi tiếp theo.

Các vé còn lại trong túi Ringo sau  $k$  vòng chơi sẽ được bỏ đi.

Với sự quan sát tinh tường, Ringo nhận ra trò chơi đã bị gian lận! Thực ra đã có một máy in được đặt trong chiếc hộp may mắn. Ở mỗi vòng chơi, người quản trò luôn chọn số nguyên  $b$  để giá trị phần thưởng của vòng chơi đó là nhỏ nhất. Giá trị được người quản trò chọn sẽ được in trên thẻ đặc biệt của vòng chơi đó.

Với những thông tin trên, Ringo muốn phân bổ các vé vào các vòng chơi của trò chơi. Cậu ấy muốn chọn tập vé để sử dụng cho mỗi vòng sao cho tổng giá trị giải thưởng của cả trò chơi là lớn nhất.

## Chi tiết cài đặt

Bài cần cài đặt hàm sau:

```
int64 find_maximum(int k, int[][] x)
```

- $k$ : số vòng chơi.

- $x$ : một mảng có kích thước  $n \times m$  thể hiện các giá trị được in trên các vé. Với mỗi màu, các vé được sắp xếp theo thứ tự không giảm của giá trị in trên vé.
- Hàm này được gọi đúng một lần.
- Hàm này cần thực hiện đúng một lời gọi hàm `allocate_tickets` (xem ở dưới), mô tả  $k$  tập vé, mỗi tập vé cho một vòng chơi. Việc phân bổ phải làm cho tổng giá trị của phần thưởng là tối đa.
- Hàm cần trả về tổng giá trị phần thưởng tối đa.

Hàm `allocate_tickets` được định nghĩa như sau:

```
void allocate_tickets(int[][] s)
```

- $s$ : một mảng có kích thước  $n \times m$ . Giá trị  $s[i][j]$  bằng  $r$  nếu vé thứ  $j$  thuộc màu  $i$  được sử dụng ở vòng thứ  $r$  của trò chơi, hoặc giá trị  $-1$  nếu vé này không được sử dụng lần nào.
- Với mỗi  $0 \leq i \leq n - 1$ , trong các phần tử  $s[i][0], s[i][1], \dots, s[i][m - 1]$  mỗi giá trị  $0, 1, 2, \dots, k - 1$  phải xuất hiện đúng một lần, còn các phần tử khác là  $-1$ .
- Nếu có nhiều phương án phân bổ cùng mang lại giá trị tổng giải thưởng tối đa, bạn có thể đưa ra bất kì phương án nào.

## Ví dụ

### Ví dụ 1

Xét lời gọi hàm sau:

```
find_maximum(2, [[0, 2, 5], [1, 1, 3]])
```

Có nghĩa là:

- có  $k = 2$  vòng chơi;
- các số nguyên được in trên các vé màu 0 lần lượt là 0, 2 và 5;
- các số nguyên được in trên các vé màu 1 lần lượt là 1, 1 và 3.

Một phương án phân bổ đạt tổng giá trị giải thưởng lớn nhất là:

- Ở vòng 0, Ringo chọn vé 0 của màu 0 (với giá trị 0) và vé 2 của màu 1 (với giá trị 3). Phần thưởng nhỏ nhất của vòng này là 3. Ví dụ, người quản trò chọn  $b = 1$ :  
 $|1 - 0| + |1 - 3| = 1 + 2 = 3$ .
- Ở vòng 1, Ringo chọn vé 2 của màu 0 (với giá trị 5) và vé 1 của màu 1 (với giá trị 1). Phần thưởng nhỏ nhất của vòng này là 4. Ví dụ, người quản trò chọn  $b = 3$ :  
 $|3 - 1| + |3 - 5| = 2 + 2 = 4$ .
- Vì vậy, tổng giá trị giải thưởng lớn nhất là  $3 + 4 = 7$ .

Để đưa ra phương án phân bổ này, hàm `find_maximum` cần phải gọi hàm `allocate_tickets` như sau:

- `allocate_tickets([[0, -1, 1], [-1, 1, 0]])`

Cuối cùng, hàm `find_maximum` cần trả về giá trị 7.

## Ví dụ 2

Xét lời gọi hàm sau:

```
find_maximum(1, [[5, 9], [1, 4], [3, 6], [2, 7]])
```

Có nghĩa là:

- chỉ có một vòng chơi;
- các số nguyên được in trên các vé màu 0 lần lượt là 5 and 9;
- các số nguyên được in trên các vé màu 1 lần lượt là 1 and 4;
- các số nguyên được in trên các vé màu 2 lần lượt là 3 and 6;
- các số nguyên được in trên các vé màu 3 lần lượt là 2 and 7.

Một phương án phân bổ đạt tổng giá trị giải thưởng lớn nhất là:

- Vòng chơi 0, Ringo chọn vé 1 của màu 0 (với giá trị 9), vé 0 của màu 1 (với giá trị 1), vé 0 của màu 2 (với giá trị 3), và vé 1 của màu 3 (với giá trị 7). Phần thưởng nhỏ nhất của vòng này là 12, khi người quản trò chọn  $b = 3$ :  
 $|3 - 9| + |3 - 1| + |3 - 3| + |3 - 7| = 6 + 2 + 0 + 4 = 12$ .

Để đưa ra phương án phân bổ này, hàm `find_maximum` cần phải gọi hàm `allocate_tickets` như sau:

- `allocate_tickets([[ -1, 0], [0, -1], [0, -1], [ -1, 0]])`

Cuối cùng, hàm `find_maximum` cần trả về giá trị 12.

## Ràng buộc

- $2 \leq n \leq 1500$  và  $n$  chẵn.
- $1 \leq k \leq m \leq 1500$
- $0 \leq x[i][j] \leq 10^9$  (với mọi  $0 \leq i \leq n - 1$  và  $0 \leq j \leq m - 1$ )
- $x[i][j - 1] \leq x[i][j]$  (với mọi  $0 \leq i \leq n - 1$  và  $1 \leq j \leq m - 1$ )

## Subtasks

1. (11 điểm)  $m = 1$
2. (16 điểm)  $k = 1$
3. (14 điểm)  $0 \leq x[i][j] \leq 1$  (với mọi  $0 \leq i \leq n - 1$  và  $0 \leq j \leq m - 1$ )
4. (14 điểm)  $k = m$
5. (12 điểm)  $n, m \leq 80$

6. (23 điểm)  $n, m \leq 300$

7. (10 điểm) Không có ràng buộc gì thêm.

## Trình chấm mẫu

Trình chấm mẫu đọc dữ liệu vào theo định dạng sau:

- dòng 1:  $n \ m \ k$
- dòng  $2 + i$  ( $0 \leq i \leq n - 1$ ):  $x[i][0] \ x[i][1] \ \dots \ x[i][m - 1]$

Trình chấm mẫu in kết quả của bạn theo định dạng sau:

- dòng 1: giá trị trả về của hàm `find_maximum`
- dòng  $2 + i$  ( $0 \leq i \leq n - 1$ ):  $s[i][0] \ s[i][1] \ \dots \ s[i][m - 1]$