Багт наадмын тасалбар (tickets)

Ринго Сингапурт багт наадамд оролцож байна. Түүний цүнхэнд шагналт тоглоомын мухлагт ашиглахыг хүсэж буй шагналын билетүүд бий. Тасалбар бүр нь n өнгийн аль нэг байх ба түүн дээр сөрөг биш бүхэл тоо хэвлэгдсэн байна. Ялгаатай тасалбарууд дээр хэвлэгдсэн тоонууд ижил байж болно. Багт наадмын хачирхалтай дүрмээс хамаарч n нь баталгаатай **тэгш** байдаг.

Рингогийн цүнхэнд өнгө бүрээс m тасалбар, нийтдээ $n \cdot m$ тасалбар байгаа. i өнгийн j тасалбар дээр $x[i][j]$ ($0 \leq i \leq n-1$ and $0 \leq j \leq m-1$) бүхэл тоо хэвлэгдсэн байдаг.

Шагналт тоглоом нь 0-ээс $k-1$ хүртэл дугаарлагдсан k үетэй. Үе бүрийг дараах дарааллаар тоглодог:

- Ринго цүнхнээсээ өнгө бүрээс нэг билет авч, n билетийн **багц** сонгоно. Тэрээр дараа нь багцаа тоглоом тоглуулагчид өгнө.
- Тоглоом тоглуулагч билетийн багц дээр хэвлэгдсэн $a[0],\ a[1]\ \ \ldots\ \ a[n-1]$ тоонуудыг хэвлэж авна. n бүхэл тоонуудын дараалал нь чухал биш байдаг.
- Тоглоом тоглуулагч шагналын хайрцгаас тусгай картыг гарган авч уг картан дээр хэвлэгдсэн b бүхэл тоог тэмдэглэж авна.
- Тоглоом тоглуулагч 0-ээс $n-1$ завсар дахь i бүрийн хувьд $a[i]$ ба b-ийн абсолют ялгаврыг тооцоолно. Абсолют ялгавруудын нийлбэрийг S гэе.
- Уг үед тоглоом тоглуугч нь S утгатай тэнцүү шагналыг Рингод өгнө.
- Уг багц тасалбаруудыг тоглоомын дараагийн үеүдэд ашиглах боломжгүй.

Уг тоглоомын k үеийн дараа Рингогийн цүнхэнд үлдсэн билетүүд хүчингүй болно.

Анхааралтай ажигласны дараа Ринго шагналт тоглоом нь луйвар болохыг ойлгожээ! Шагналын хайрцаг дотор үнэхээр принтер байдаг. Үе бүрд тухайн үеийн шагналын утгыг хамгийн бага байлгах b бүхэл тоог тоглоом тоглуулагч олно. Тоглоом тоглуулагч тооцоолсон утгаа энэ үеийн тусгай карт дээр хэвлэдэг.

Энэ бүх мэдээллийг авсан Ринго тоглоомын үеүдэд тасалбарыг хуваарилахыг хүсэж байна. Үүний тулд тэрээр шагналын нийт утга хамгийн их байлгахын тулд үе бүрд ашиглах тасалбарын багцыг сонгохыг хүсэж байна.

Хэрэгжүүлэлтийн мэдээлэл

Та дараах функцийг хэрэгжүүлэх ёстой:

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

- k: үеийн тоо.
- x: $n \times m$ тасалбар бүрийг тодорхойлох бүхэл утгатай массив. Өнгө тус бүрийн тасалбарын бүхэл утгуудыг үл буурах дараалаар эрэмбэлсэн.
- Уг функц нь зөвхөн ганц удаа дуудагдана.
- Энэхүү функц нь `allocate_tickets` (дор тодорхойлсоныг харна уу) функцийг цор ганц удаа дуудаж, үе бүрд зориулан k тасалбарын багцыг тодорхойлно. Уг хуваарилалт нь шагналын нийт утгыг хамгийн их байлгах ёстой.
- Уг функц нь шагналын нийт утгын хамгийн ихийг буцаах болно.

`allocate_tickets` функц дараах байдлаар тодорхойлогдоно:

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

- s: $n \times m$ массив. Хэрэв i өнгөтэй j тасалбар нь тоглоомын r үеийн багцад хэрэглэгдэх бол $s[i][j]$-ийн утга r, хэрэглэгдэхгүй бол -1 байна.
- $0 \leq i \leq n-1$ бүрийн хувьд $s[i][0], s[i][1], \ldots, s[i][m-1]$ утгуудын дотор $0, 1, 2, \ldots, k-1$ утгууд нь зөвхөн ганц удаа байх ба бусад оролтууд нь -1 байх ёстой.
- Хэрэв нийт шагналын хамгийн их хэмжээнд хүрч болох хэд хэдэн хуваарилалтууд байгаа бол тэдгээрийн аль нэгийг мэдээлэхийг зөвшөөрнө.

Жишээ

Жишээ 1

Дараах дуудалтыг хийсэн гэе:

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

Энэ дуудалт нь:

- $k = 2$ үетэй;
- 0 өнгийн тасалбар тус бүр дээр 0, 2 5 бүхэл тоонууд хэвлэгдсэн
- 1 өнгийн тасалбар тус бүр дээр 1, 1, 3 бүхэл тоонууд хэвлэгдсэн.

Нийт шагналын хамгийн их утгыг өгөх боломжтой хуваарилалт нь:

- Тоглоомын 0 дүгээр үед Ринго 0 өнгийн 0 дүгээр (0 утгатай), 1 өнгийн 2 дугаар (3 утгатай) тасалбар авна. Энэ үеийн шагналын боломжит хамгийн бага утга нь 3. Тухайлбал, тоглоом тоглуулагч нь $b = 1$ гэж сонгосноор $b = 1$: $|1-0| + |1-3| = 1 + 2 = 3$ болно.
- Тоглоомын 1 дүгээр үед, Ринго 0 өнгийн 2 дугаар (5 утгатай), 1 өнгийн 1 дүгээр (1 утгатай) тасалбарыг сонгоно. Энэ үеийн шагналын боломжит хамгийн бага утга нь 4. Тухайлбал, тоглоом тоглуулагч нь $b = 3$ гэж авснаар $b = 3$: $|3-1| + |3-5| = 2 + 2 = 4$.
- Тиймээс шагналын нийт утга нь $3 + 4 = 7$.

Энэхүү хуваарилалтыг хийхийн тулд `find_maximum` функц нь `allocate_tickets` функцийг дараах байдлаар дуудна:

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

Эцэст нь `find_maximum` функц нь 7 утга буцаах болно.

Жишээ 2

Дараах дуудалтыг авч үзье:

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

Энэ дуудалт нь:

- Зөвхөн ганц үетэй,
- 0 өнгийн тасалбар тус бүр дээр 5 ба 9 тоонууд хэвлэгдсэн;
- 1 өнгийн тасалбар тус бүр дээр 1 ба 4 бүхэл тоонууд хэвлэгдсэн;
- 2 өнгийн тасалбар тус бүр дээр 3 ба 6 бүхэл тоонууд хэвлэгдсэн;
- 3 өнгийн тасалбар тус бүр дээр 2 ба 7 тоонууд хэвлэгдсэн.

Нийт шагналын хамгийн их утгыг өгөх боломжтой хуваарилалт нь:

- 0 дугаар үед Ринго 0 өнгийн 1 дүгээр (9 утгатай), 1 өнгийн 0 дүгээр (1 утгатай), 2 өнгийн 0 дүгээр (3 утгатай) болон 3 өнгийн 1 дүгээр (7 утгатай) тасалбарыг сонгоно. Энэ үеийн шагналын боломжит хамгийн бага утга нь 12. Тоглоом тоглуулагч $b = 3$ гэж сонгосноор $b = 3$: $|3-9| + |3-1| + |3-3| + |3-7| = 6 + 2 + 0 + 4 = 12$.

Энэхүү шийдлийг гаргахын тулд `find_maximum` функц нь `allocate_tickets` функцийг дараах байдлаар дуудна:

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

Эцэст нь `find_maximum` функц 12 утгыг буцаах болно.

Хязгаарлалт

- $2 \le n \le 1500$ ба n нь тэгш тоо.
- $1 \le k \le m \le 1500$
- $0 \le x[i][j] \le 10^9$ (бүх $0 \le i \le n-1$ ба $0 \le j \le m-1$ утгын хувьд)
- $x[i][j-1] \le x[i][j]$ (бүх $0 \le i \le n-1$ ба $1 \le j \le m-1$ утгын хувьд)

Дэд бодлого

1. (11 оноо) $m = 1$
2. (16 оноо) $k = 1$

3. (14 оноо) $0 \leq x[i][j] \leq 1$ (бүх $0 \leq i \leq n-1$ and $0 \leq j \leq m-1$ утгын хувьд)
4. (14 оноо) $k = m$
5. (12 оноо) $n, m \leq 80$
6. (23 оноо) $n, m \leq 300$
7. (10 оноо) Нэмэлт хязгаарлалт байхгүй.

Жишээ шалгагч

Жишээ шалгагч нь дараах форматтай оролтыг уншина:

- мөр 1: $\;n\;\;m\;\;k$
- мөр $2+i$ ($0 \leq i \leq n-1$): $\;x[i][0]\;\;x[i][1]\;\;\ldots\;\;x[i][m-1]$

Жишээ шалгагч нь таны хариултыг дараах форматаар хэвлэнэ:

- мөр 1: `find_maximum` функцийн буцаасан утга
- мөр $2+i$ ($0 \leq i \leq n-1$): $\;s[i][0]\;\;s[i][1]\;\;\ldots\;\;s[i][m-1]$