Zonal Computing Olympiad, 2024

Question paper

Vegetables

You are a farmer, and you want to grow a wide variety of vegetables so that the people in your town can eat a balanced diet.

In order to remain healthy, a person must eat a diet that contains N essential vegetables, numbered from 1 to N. In total, your town requires A_i units of each vegetable i, for $1 \le i \le N$. In order to grow a **single** unit of vegetable i, you require B_i units of water.

However, you can use **upgrades** to improve the efficiency of your farm. In a single upgrade, you can do **one** of the following two actions:

1. You can improve the nutritional value of your produce so that your town requires one less unit of some vegetable i. Specifically, you can choose any one vegetable i such that $A_i \ge 1$, and reduce A_i by 1.
2. You can improve the quality of your soil so that growing one unit of some vegetable i requires one less unit of water. Specifically, you can choose any one vegetable i such that $B_i \ge 1$, and reduce B_i by 1.

You wish to answer Q questions numbered from 1 to Q, where the j-th question is the following:

- If you use at most X_j upgrades, what is the **minimum** possible number of units of water you will need to feed your town?

Input Format

The first line contains two space-separated integers N and Q, the number of essential vegetables and the number of questions, respectively.

The second line contains N space separated integers, $A_1\ A_2\ \ldots\ A_N$.

The third line contains N space separated integers, $B_1\ B_2\ \ldots\ B_N$.

The following Q lines describe the questions. The j-th of these lines contains a single integer X_j.

Output Format

You should print Q lines of output. The j-th line should be the answer to the j-th question.

Scoring

The test data for this problem is divided into multiple subtasks. In order to pass a subtask, your submitted program must solve every test case within that subtask correctly and within the time and memory limits.

You will be awarded the points allocated to a subtask if **at least one** submission you make during the contest passes that subtask. You do **not** need to combine your solutions for multiple subtasks into a single submission.

Please keep in mind that the subtasks are not necessarily arranged in increasing order of difficulty. We encourage you to try as many subtasks as possible.

Constraints

In all test data, it is guaranteed that:

- $1 \le N \le 2 * 10^5$.
- $1 \le Q \le 2 * 10^5$.
- $1 \le A_i \le 10^6$ for all $1 \le i \le N$.
- $1 \le B_i \le 10^6$ for all $1 \le i \le N$.
- $1 \le X_j \le 10^9$ for all $1 \le j \le Q$.

Please be aware that the output for this problem may not fit in 32-bit integers. You may need to use 64-bit integers in your computations.

Subtasks

- **Subtask 1** (7 points) $Q = 1$, $X_j = 1$, $N \le 10$, $A_i \le 10^3$, $B_i \le 10^3$.
- **Subtask 2** (6 points) $Q \le 3$, $X_j \le 3$, $N \le 10$.
- **Subtask 3** (9 points) $A_i = 1$, $N \le 10^3$, $Q \le 10^3$.
- **Subtask 4** (7 points) $A_i = 1$.
- **Subtask 5** (18 points) $N, Q, X_j \le 30$.
- **Subtask 6** (15 points) $N, Q, X_j \le 400$.
- **Subtask 7** (16 points) $N, Q \le 10^3$.
- **Subtask 8** (7 points) The sum of all A_i does not exceed $2 * 10^5$. Also, the sum of all B_i does not exceed $2 * 10^5$.
- **Subtask 9** (7 points) $X_j \le 2 * 10^5$.
- **Subtask 10** (8 points) No additional constraints.

Sample 0

Input

```
4 2
2 4 5 3
5 2 3 3
1
2
```

Output

```
37
32
```

Explanation

In sample 0, there are two questions.

- For the first question, $X_1 = 1$ so you are allowed up to 1 upgrade. One optimal choice is to reduce A_1 by 1. Then, the final array A is $[1, 4, 5, 3]$ and the final array B is $[5, 2, 3, 3]$. This yields a water requirement of 37 units.
- For the second question, $X_2 = 2$ so you are allowed up to 2 upgrades. One optimal first move is to reduce B_3 by 1. Then, an optimal second move is to reduce A_1 by 1. Then, the final array A is $[1, 4, 5, 3]$ and the final array B is $[5, 2, 2, 3]$. This yields a water requirement of 32 units.

Subtask Validity

Sample 0 is valid for subtasks 2, 5, 6, 7, 8, 9 and 10.

Sample 1

Input

```
4 1
1 4 2 3
5 4 3 6
29
```

Output

```
0
```

Explanation

In sample 1, there is only one question. You are allowed up to `29` upgrades. You can use `28` upgrades to make the final array `A` and the final array `B` both `[0, 0, 0, 0]`. This yields a water requirement of `0` units. Note that you are not required to use all `29` upgrades.

Subtask Validity

Sample 1 is valid for subtasks 5, 6, 7, 8, 9 and 10.

Sample 2

Input

```
14 4
1 1 1 1 1 1 1 1 1 1 1 1 1 1
4 2 4 12 10 5 2 6 4 10 2 3 5 12
3
9
5
6
```

Output

```
47
13
31
26
```

Subtask Validity

Sample 2 is valid for subtasks 3, 4, 5, 6, 7, 8, 9 and 10.

Sample 3

Input

```
8 6
7 9 4 2 2 2 7 7
1 10 6 7 4 6 8 4
29
3
12
27
8
19
```

Output

```
8
209
125
20
159
72
```

Subtask Validity

Sample 3 is valid for subtasks 5, 6, 7, 8, 9 and 10.

Sample 4

Input

```
1 1
1000000
1000000
1
```

Output

```
999999000000
```

Subtask Validity

Sample 4 is valid for subtasks 2, 5, 6, 7, 9 and 10.

Fruits

You are still a farmer. Now that you are tired of growing vegetables, you have decided to start growing fruits. Your farm has $N * M$ **fields** arranged in N rows numbered from 1 to N and M columns numbered from 1 to M. The field at the i-th row and j-th column is denoted by (i, j). Your farm grows K distinct **fruits**, numbered from 1 to K. Each field (i, j) grows a single plant that bears the fruit $A_{i,j}$.

Your farm is home to many **bees**. A bee can move from a field (i, j) to one of the fields $(i-1, j)$, $(i+1, j)$, $(i, j-1)$, or $(i, j+1)$ in one second (assuming that field exists). Bees carry pollen from one plant to another, which is critical for the production of fruits. Whenever a bee travels from some plant (x_1, y_1) to some other plant (x_2, y_2), it chooses a route such that it takes the **smallest** amount of time possible.

In order to study the effect of pollination in your farm as you choose which plants to grow, you would like to perform Q operations numbered from 1 to Q. Each operation is either **type 1** or **type 2**.

Each operation q (such that $1 \le q \le Q$) contains a number T, which specifies the type of the operation.

- If $T = 1$, you are given a field (I, J) and a fruit X, and asked to change the fruit produced by the field (I, J) to X.
- If $T = 2$, you are given two distinct fruits U and V. Your task is to find a field (x_1, y_1) that grows fruit U and a field (x_2, y_2) that grows fruit V such that a bee carrying pollen from (x_1, y_1) to (x_2, y_2) will take the **largest** amount of time possible, and report this time in seconds. It is guaranteed that at any time, there will be at least one instance of each of the K fruits.

Note that just before an operation of type 1, the field (I, J) could have been producing fruit X. That is, it is possible for no change to occur during an operation of type 1.

Input Format

The first line contains three space-separated integers N, M and K, the number of rows in the farm, the number of columns in the farm, and the different fruits that can grow on the farm.

Each of the next N lines contain M space separated integers. The i-th of these lines contains $A_{i,1}$ $A_{i,2}$ ... $A_{i,M}$, denoting the fruits growing at fields $(i, 1)$, $(i, 2)$, ... (i, M) respectively.

The next line contains a single integer Q, denoting the number of operations.

The next Q lines contain information about the operations. The q-th line describes the q-th operation. The first number in the line will be T, the type of the q-th operation.

- If $T = 1$, the line will further contain three integers I, J and X.
- If $T = 2$, the line will further contain two integers U and V.

Output Format

For each operation of type 2, output a single integer, the answer to the operation, on a new line.

Scoring

The test data for this problem is divided into multiple subtasks. In order to pass a subtask, your submitted program must solve every test case within that subtask correctly and within the time and memory limits.

You will be awarded the points allocated to a subtask if **at least one** submission you make during the contest passes that subtask. You do **not** need to combine your solutions for multiple subtasks into a single

submission.

Please keep in mind that the subtasks are not necessarily arranged in increasing order of difficulty. We encourage you to try as many subtasks as possible.

Constraints

In all test data, it is guaranteed that:

- $1 \le N \le 3 * 10^5$
- $1 \le M \le 3 * 10^5$
- $2 \le N * M \le 3 * 10^5$
- $2 \le K \le N * M$
- $1 \le A_{i,j} \le K$ for all $1 \le i \le N$ and $1 \le j \le M$
- $1 \le Q \le 2 * 10^5$
- For each q such that $1 \le q \le Q$:
  - $T = 1$ or $T = 2$.
  - If $T = 1$, then $1 \le I \le N$, $1 \le J \le M$, $1 \le X \le K$.
  - If $T = 2$, then $1 \le U \le K$, $1 \le V \le K$. Furthermore, $U \ne V$.
- At any time, there will be at least one instance of each of the K fruits.
- There is at least one operation of type 2 in the input.

Subtasks

- **Subtask 1** (7 points) $Q = 1$, $N * M \le 100$.
- **Subtask 2** (8 points) $Q = 1$, $N * M \le 500$.
- **Subtask 3** (13 points) $Q = 1$, $N * M \le 2 * 10^3$.
- **Subtask 4** (18 points) $Q = 1$.
- **Subtask 5** (12 points) $N = 1$ and there will be no operations of type 1.
- **Subtask 6** (9 points) $N = 1$.
- **Subtask 7** (13 points) $N \le 5$ and there will be no operations of type 1.
- **Subtask 8** (12 points) There will be no operations of type 1.
- **Subtask 9** (8 points) No additional constraints.

Sample 0

Input

```
5 5 9
7 1 5 5 4
2 4 6 2 8
7 1 3 5 6
9 4 4 8 2
1 3 6 9 3
1
2 1 3
```

Output

```
7
```

Explanation

There is only one operation of type 2, with `U = 1` and `V = 3`. The fields containing fruit `1` are `(1, 2)`, `(3, 2)`, `(5, 1)`. Similarly, the fields containing fruit `3` are `(3, 3)`, `(5, 2)`, `(5, 5)`.

In order to maximize the time taken by the bee, it is optimal to choose the fields `(1, 2)` and `(5, 5)`. One possibility for the quickest path that a bee can take between these fields is: `(1, 2) (2, 2) (3, 2) (4, 2) (4, 3) (4, 4) (4, 5) (5, 5)`. This path takes `7` seconds, so the answer for this operation is `7`.

It is not possible to find a quicker path between `(1, 2)` and `(5, 5)`. Further, if any other pair of fields is chosen, the quickest path between that pair will take time less than or equal to `7` seconds.

Subtask Validity

Sample 0 is valid for subtasks 1, 2, 3, 4, 7, 8 and 9.

Sample 1

Input

```
4 3 12
8 9 1
3 10 12
2 5 11
4 6 7
7
2 1 6
2 6 9
2 2 11
1 3 3 11
2 5 8
2 8 2
2 2 11
```

Output

```
4
3
2
3
2
2
```

Subtask Validity

Sample 1 is valid for subtask 9.

Sample 2

Input

```
1 9 4
2 1 1 1 3 4 2 3 3
6
2 1 2
2 3 2
2 1 3
2 2 4
2 4 1
2 3 4
```

Output

```
5
8
7
5
4
3
```

Subtask Validity

Sample 2 is valid for subtasks 5, 6, 7, 8 and 9.

Sample 3

Input

```
1 13 7
6 4 3 4 5 1 7 7 1 6 7 2 4
5
2 7 3
2 3 4
1 1 2 3
2 4 7
2 4 3
```

Output

```
8
10
7
11
```

Subtask Validity

Sample 3 is valid for subtasks 6 and 9.

Sample 4

Input

```
 7 7 20
9 17 15 17 18 20 11
7 12 13 18 12 16 18
3 18 6 8 5 10 16
7 11 5 17 2 2 1
8 11 4 12 3 2 12
19 2 7 13 3 6 14
6 8 8 15 9 20 17
6
2 17 8
2 1 12
2 6 14
2 17 1
2 18 8
2 2 8
```

Output

```
 8
7
7
8
10
7
```

Subtask Validity

Sample 4 is valid for subtasks 8 and 9.