Public Transport (publictransport)

Cairo City is a metropolis with $N \cdot M$ public transport stations, where $N, M \geq 3$. The stations are arranged in N concentric circles, and there are M stations over each circle. The circles are numbered from 0 to $N-1$, and for each circle, the stations are numbered from 0 to $M-1$, inclusive. We refer to station j on circle i as station (i, j) (where $i = 0 \ldots N-1$, $j = 0 \ldots M-1$).

Figure 1: Citizens of Cairo City in need of efficient public transport.

In the city, there are N tram lines. Tram lines are bidirectional and periodic, each connecting consecutive stations of some circle. Let tram line k (where $k = 0 \ldots N-1$) connect the stations on circle k, i.e., stations $(k, 0) \ldots (k, M-1)$. Station $(k, M-1)$ is connected with station $(k, 0)$.

Additionally, there are M metro lines in the city. Metro lines are bidirectional, each connecting consecutive stations at the same position over different circles. Let metro line l (where $l = 0 \ldots M-1$) connect stations at position l on each circle, i.e., stations $(0, l) \ldots (N-1, l)$. Stations $(N-1, l)$ and$(0, l)$ are not connected.

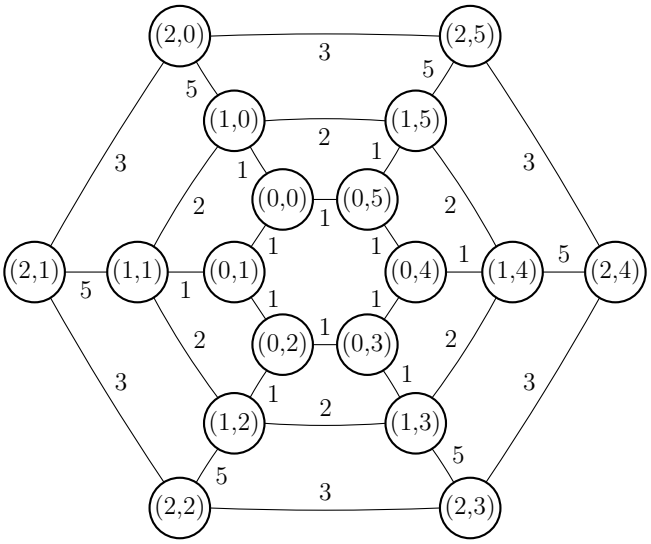

Figure 2: Transport network for $N = 3$ and $M = 6$, having $S_0 = 1, S_1 = 2, S_2 = 3$ and $T_0 = 1, T_1 = 5$.

Traveling between consecutive stations via tram line k takes S_k minutes (where $k = 0 \ldots N-1$). For each metro line l (where $l = 0 \ldots M-1$), traveling between stations (i, l) and $(i+1, l)$ takes T_i minutes (where $i = 0 \ldots N-2$).

Your task is to determine shortest travel times in this public transport network. You are given Q queries, each query specifies two public transport stations (A_q, B_q) and (C_q, D_q) (where $q = 0 \ldots Q-1$). For each query, find the minimum amount of time it takes to get from the first station to the second one, assuming that changing lines doesn't take extra time.

> ☞ Among the attachments of this task you may find a template file `publictransport.*` with a sample incomplete implementation.

Input

The first line contains integers N, M and Q. The second line contains N integers S_k. The third line contains $N-1$ integers T_i.

Each of the next Q lines contains four integers A_q, B_q, C_q and D_q.

Output

For each query, output a single number on a separate line, the minimum travel time (in minutes) between the specified stations.

Constraints

- $3 \leq N \leq 100\,000$.
- $3 \leq M \leq 100\,000$.
- $1 \leq Q \leq 200\,000$.
- $1 \leq S_k \leq 1\,000\,000\,000$ for each $k = 0 \ldots N-1$.
- $1 \leq T_i \leq 1\,000\,000\,000$ for each $i = 0 \ldots N-2$.
- $0 \leq A_q, C_q < N$ and $0 \leq B_q, D_q < M$ for each $q = 0 \ldots Q-1$.
- $(A_q, B_q) \neq (C_q, D_q)$ for each $q = 0 \ldots Q-1$.

Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points) — Example.

– **Subtask 2** (16 points) — $N, M \leq 50$.

– **Subtask 3** (13 points) — $M = 3$ and $N \leq 2\,000$.

– **Subtask 4** (15 points) — $N \cdot M \leq 500\,000$, all T_i-s are equal and $S_{k-1} \leq S_k$ for each $1 \leq k < N$.

– **Subtask 5** (23 points) — All T_i-s are equal and $S_{k-1} \leq S_k$ for each $1 \leq k < N$.

– **Subtask 6** (10 points) $S_{k-1} \leq S_k$ for each $1 \leq k < N$.

– **Subtask 7** (23 points) No additional limitations.

Example

input	output
3 6 2 1 2 3 1 5 1 2 1 5 2 1 2 4	5 9

Explanation

The public transport network in the **sample case** is displayed in Figure 2.

In the first query, the minimum time to reach station $(1, 5)$ from $(1, 2)$ is 5 minutes, e.g., by taking the path $(1, 2) - (0, 2) - (0, 3) - (0, 4) - (0, 5) - (1, 5)$.

In the second query, the minimum time to reach station $(2, 4)$ from $(2, 1)$ is 9 minutes, e.g., by taking the path $(2, 1) - (2, 0) - (2, 5) - (2, 4)$.