A - A Substring

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 100 points

Problem Statement

You are given an N-character string S consisting of lowercase English letters.

Output the string of $N-A-B$ characters obtained by removing the first A characters and the last B characters from S.

Constraints

- $1 \leq N \leq 20$
- $0 \leq A$
- $0 \leq B$
- $A+B < N$
- S is a string of N characters consisting of lowercase English letters.
- N, A, and B are all integers.

Input

The input is given from Standard Input in the following format:

$$N \quad A \quad B$$
$$S$$

Output

Output the string obtained by removing the first A characters and the last B characters from S.

Sample Input 1

```
7 1 3
atcoder
```

Sample Output 1

```
tco
```

Removing the first one character and the last three characters from `atcoder` gives `tco`.

Sample Input 2

```
1 0 0
a
```

Sample Output 2

```
a
```

The string to be output may be equal to S.

Sample Input 3

```
20 4 8
abcdefghijklmnopqrst
```

Sample Output 3

```
efghijkl
```

B - Search and Delete

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 200 points

Problem Statement

Takahashi has an integer sequence $A = (A_1, A_2, \ldots, A_N)$ of length N.
It is guaranteed that A is non-decreasing.

Takahashi performs M operations on this integer sequence. In the i-th operation $(1 \leq i \leq M)$, he performs the following operation:

> If the sequence A contains B_i as an element, select one such element and delete it. If no such element exists, do nothing.
>
> Note that since A is non-decreasing, the sequence after the operation is uniquely determined regardless of the choice of element, and remains non-decreasing.

Find A after performing M operations.

▶ What is non-decreasing?

Constraints

- $1 \leq N \leq 100$
- $1 \leq M \leq 100$
- $1 \leq A_i, B_i \leq 10^9$
- The integer sequence A is non-decreasing.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad M \\
A_1 \quad A_2 \quad \ldots \quad A_N \\
B_1 \quad B_2 \quad \ldots \quad B_M
\end{array}
$$

Output

Output the elements of A after the operations, in order, separated by spaces on a single line.
If A is empty after the operations, output nothing.

Sample Input 1

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

Sample Output 1

```
1 3 3 5 6
```

Initially, A is $A = (1, 2, 2, 3, 3, 3, 5, 6)$.
The operations are performed as follows:

- Delete one 2 from A, and A becomes $A = (1, 2, 3, 3, 3, 5, 6)$ after the operation.
- Delete one 2 from A, and A becomes $A = (1, 3, 3, 3, 5, 6)$ after the operation.
- Since A does not contain 7 as an element, do nothing. A remains $A = (1, 3, 3, 3, 5, 6)$ after the operation.
- Delete one 3 from A, and A becomes $A = (1, 3, 3, 5, 6)$ after the operation.
- Since A does not contain 2 as an element, do nothing. A remains $A = (1, 3, 3, 5, 6)$ after the operation.

Therefore, after all operations, $A = (1, 3, 3, 5, 6)$, so output the elements in this order separated by spaces.

Sample Input 2

```
1 2
1
1 1
```

Sample Output 2

```

```

A becomes empty after the operations, so output nothing.

C - Distance Indicators

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 300 points

Problem Statement

You are given an integer sequence $A = (A_1, A_2, \dots, A_N)$ of length N.

Find how many pairs of integers (i, j) $(1 \leq i < j \leq N)$ satisfy $j - i = A_i + A_j$.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq A_i \leq 2 \times 10^5$ $(1 \leq i \leq N)$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

N
A_1 A_2 $\dots$ A_N

Output

Output the answer.

Sample Input 1

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

Sample Output 1

```
3
```

For example, when $(i, j) = (4, 7)$, we have $j - i = 7 - 4 = 3$ and $A_i + A_j = 1 + 2 = 3$, so $j - i = A_i + A_j$.

In contrast, when $(i, j) = (3, 8)$, we have $j - i = 8 - 3 = 5$ and $A_i + A_j = 4 + 6 = 10$, so $j - i \neq A_i + A_j$.

Only the three pairs $(i, j) = (1, 9), (2, 4), (4, 7)$ satisfy the condition, so output `3`.

Sample Input 2

```
3
123456 123456 123456
```

Sample Output 2

```
0
```

There may be no pairs that satisfy the condition.

Sample Input 3

```
30
8 3 6 4 9 6 5 6 5 6 3 4 7 3 7 4 9 8 5 8 3 6 8 8 4 5 5 5 6 5
```

Sample Output 3

```
17
```

D - Takahashi's Expectation

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 425 points

Problem Statement

Takahashi will receive N presents.

He has a parameter called mood, which is a non-negative integer, and his mood changes every time he receives a present. Each present has three parameters: value P, mood increase A, and mood decrease B, and his mood changes as follows based on these parameters:

- When the value P of the received present is greater than or equal to his mood, he is happy with the present, and his mood increases by A.
- When the value P of the received present is less than his mood, he is disappointed with the present, and his mood decreases by B. However, if his mood is originally less than B, it becomes 0.

The i-th $(1 \leq i \leq N)$ present he receives has value P_i, mood increase A_i, and mood decrease B_i.

You are given Q questions, so answer all of them. In the i-th $(1 \leq i \leq Q)$ question, you are given a non-negative integer X_i, so answer the following question:

> Find Takahashi's mood after receiving all N presents when his mood is initially X_i.

Constraints

- $1 \leq N \leq 10000$
- $1 \leq P_i \leq 500 \ (1 \leq i \leq N)$
- $1 \leq A_i \leq 500 \ (1 \leq i \leq N)$
- $1 \leq B_i \leq 500 \ (1 \leq i \leq N)$
- $1 \leq Q \leq 5 \times 10^5$
- $0 \leq X_i \leq 10^9 \ (1 \leq i \leq Q)$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
P_1 \quad A_1 \quad B_1 \\
P_2 \quad A_2 \quad B_2 \\
\vdots \\
P_N \quad A_N \quad B_N \\
Q \\
X_1 \\
X_2 \\
\vdots \\
X_Q
\end{array}
$$

Output

Output Q lines. The i-th line should contain the answer to the i-th question.

Sample Input 1

```
4
3 1 4
1 5 9
2 6 5
3 5 8
11
0
1
2
3
4
5
6
7
8
9
10
```

Sample Output 1

```
6
0
0
0
5
6
0
0
0
0
0
```

When Takahashi's initial mood is 10, his mood changes as follows:

- The value 3 of the first present is less than his mood 10, so his mood decreases by the mood decrease 4, and his mood becomes 6.
- The value 1 of the second present is less than his mood 6, and Takahashi's mood 6 is less than the mood decrease 9, so his mood becomes 0.
- The value 2 of the third present is not less than his mood 0, so his mood increases by the mood increase 6, and his mood becomes 6.
- The value 3 of the fourth present is less than his mood 6, and Takahashi's mood 6 is less than the mood decrease 8, so his mood becomes 0.

Therefore, his final mood is 0.

Sample Input 2

```
3
500 500 500
500 500 500
500 500 500
1
1000000000
```

Sample Output 2

```
999998500
```

Because Takahashi's mood is too high, his mood keeps decreasing even when he receives the best presents.

Sample Input 3

```
20
124 370 105
280 200 420
425 204 302
435 141 334
212 287 231
262 410 481
227 388 466
222 314 366
307 205 401
226 460 452
336 291 119
302 104 432
478 348 292
246 337 403
102 404 371
368 399 417
291 416 351
236 263 231
170 415 482
101 339 184
20
1162
1394
1695
2501
3008
3298
4053
4093
4330
5199
5302
5869
5875
6332
6567
7483
7562
7725
9723
9845
```

Sample Output 3

```
339
339
339
339
339
339
339
339
339
339
339
339
339
389
339
643
722
885
2883
3005
```

E - A Path in A Dictionary

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 475 points

Problem Statement

You are given a simple connected undirected graph G with N vertices and M edges.
The vertices of G are numbered vertex 1, vertex $2, \ldots$, vertex N, and the i-th $(1 \leq i \leq M)$ edge connects vertices U_i and V_i.

Find the lexicographically smallest simple path from vertex X to vertex Y in G.
That is, find the lexicographically smallest among the integer sequences $P = (P_1, P_2, \ldots, P_{|P|})$ that satisfy the following conditions:

- $1 \leq P_i \leq N$
- If $i \neq j$, then $P_i \neq P_j$.
- $P_1 = X$ and $P_{|P|} = Y$.
- For $1 \leq i \leq |P| - 1$, there exists an edge connecting vertices P_i and P_{i+1}.

One can prove that such a path always exists under the constraints of this problem.

You are given T test cases, so find the answer for each.

▶ Lexicographic order on integer sequences

Constraints

- $1 \leq T \leq 500$
- $2 \leq N \leq 1000$
- $N - 1 \leq M \leq \min\left(\frac{N(N-1)}{2}, 5 \times 10^4\right)$
- $1 \leq X, Y \leq N$
- $X \neq Y$
- $1 \leq U_i < V_i \leq N$
- If $i \neq j$, then $(U_i, V_i) \neq (U_j, V_j)$.
- The given graph is connected.
- The sum of N over all test cases in each input is at most 1000.
- The sum of M over all test cases in each input is at most 5×10^4.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
T \\
\mathrm{case}_1 \\
\mathrm{case}_2 \\
\vdots \\
\mathrm{case}_T
\end{array}
$$

case_i represents the i-th test case. Each test case is given in the following format:

$$
\begin{array}{l}
N \quad M \quad X \quad Y \\
U_1 \quad V_1 \\
U_2 \quad V_2 \\
\vdots \\
U_M \quad V_M
\end{array}
$$

Output

Output T lines.

The i-th line $(1 \leq i \leq T)$ should contain the vertex numbers on the simple path that is the answer to the i-th test case, in order, separated by spaces.

That is, when the answer to the i-th test case is $P = (P_1, P_2, \ldots, P_{|P|})$, output $P_1, P_2, \ldots, P_{|P|}$ on the i-th line in this order, separated by spaces.

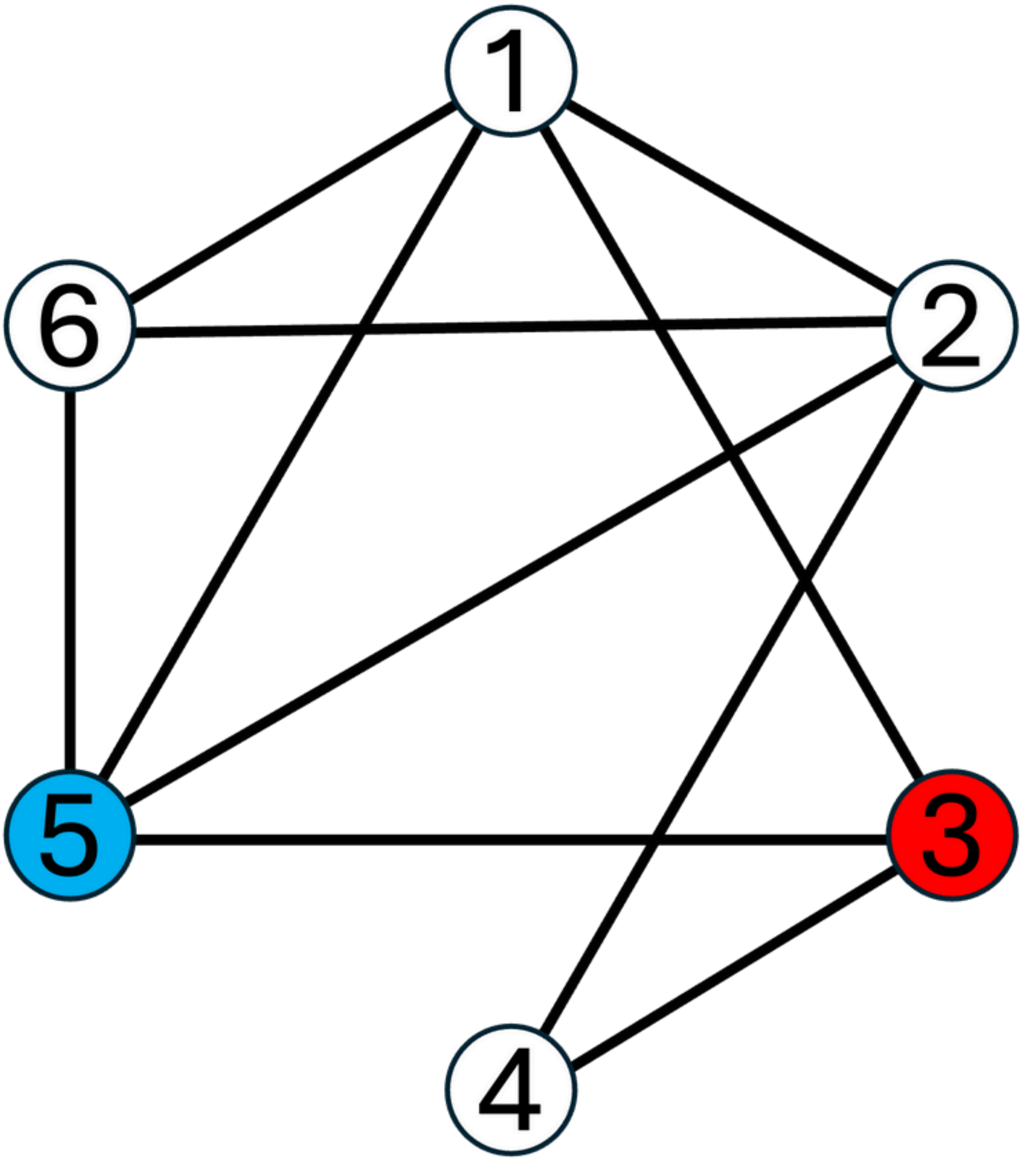
Sample Input 1

```
2
6 10 3 5
1 2
1 3
1 5
1 6
2 4
2 5
2 6
3 4
3 5
5 6
3 2 3 2
1 3
2 3
```

Sample Output 1

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

For the first test case, graph G is as follows:



The simple paths from vertex 3 to vertex 5 on G, listed in lexicographic order, are as follows:

- $(3, 1, 2, 5)$
- $(3, 1, 2, 6, 5)$
- $(3, 1, 5)$
- $(3, 1, 6, 2, 5)$
- $(3, 1, 6, 5)$
- $(3, 4, 2, 1, 5)$
- $(3, 4, 2, 1, 6, 5)$
- $(3, 4, 2, 5)$
- $(3, 4, 2, 6, 1, 5)$
- $(3, 4, 2, 6, 5)$
- $(3, 5)$

Among these, the lexicographically smallest is $(3, 1, 2, 5)$, so output $3, 1, 2, 5$ separated by spaces on the first line.

For the second test case, $(3, 2)$ is the only simple path from vertex 3 to vertex 2.

F - Random Gathering

Time Limit: 3 sec / Memory Limit: 1024 MiB

Score : 500 points

Problem Statement

There are N plates arranged from left to right as plate 1, plate $2, \ldots,$ plate N. Initially, plate i $(1 \leq i \leq N)$ contains A_i stones.

You will perform M operations on these plates. In the i-th operation $(1 \leq i \leq M)$, two integers L_i and R_i are given, and the following operations are performed in order:

- Remove all stones from the $R_i - L_i + 1$ plates: plate L_i, plate $L_i + 1, \ldots,$ plate R_i.
- Uniformly randomly choose an integer between L_i and R_i, inclusive, and let it be x.
- Place all the removed stones on plate x.

For $i = 1, 2, \ldots, N$, find the expected number, modulo 998244353, of stones placed on plate i when all M operations are completed.

▶ Finding expected value modulo 998244353

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq M \leq 2 \times 10^5$
- $0 \leq A_i < 998244353$ $(1 \leq i \leq N)$
- $1 \leq L_i \leq R_i \leq N$ $(1 \leq i \leq M)$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad M \\
A_1 \quad A_2 \quad \ldots \quad A_N \\
L_1 \quad R_1 \\
L_2 \quad R_2 \\
\vdots \\
L_M \quad R_M
\end{array}
$$

Output

Output N integers separated by spaces on a single line. For the i-th $(1 \leq i \leq N)$, find the expected number, modulo 998244353, of stones placed on plate i when all M operations are completed, and output it.

Sample Input 1

```
7 4
30 10 40 10 50 90 20
4 6
5 7
1 6
3 7
```

Sample Output 1

```
35 35 36 36 36 36 36
```

For example, the operations proceed as follows:

- In the first operation, 4 is chosen. The number of stones on plates $1, 2, \ldots, 7$ becomes $30, 10, 40, 150, 0, 0, 20$, respectively.
- In the second operation, 6 is chosen. The number of stones on plates $1, 2, \ldots, 7$ becomes $30, 10, 40, 150, 0, 20, 0$, respectively.
- In the third operation, 2 is chosen. The number of stones on plates $1, 2, \ldots, 7$ becomes $0, 250, 0, 0, 0, 0, 0$, respectively.
- In the fourth operation, 3 is chosen. The number of stones on plates $1, 2, \ldots, 7$ becomes $0, 250, 0, 0, 0, 0, 0$, respectively.

When all operations are completed, the expected number of stones on plates $1, 2$ is 35, and the expected number of stones on plates $3, 4, 5, 6, 7$ is 36, so output `35 35 36 36 36 36 36`.

Sample Input 2

```
2 1
0 1
1 2
```

Sample Output 2

```
499122177 499122177
```

Note that you need to find the expected value modulo 998244353.

When all operations are completed, for both plates, there is a $\frac{1}{2}$ probability that one stone is placed, and a $\frac{1}{2}$ probability that no stone is placed. Therefore, the expected number of stones placed is $\frac{1}{2}$. We have $499122177 \times 2 \equiv 1 \pmod{998244353}$, so output `499122177 499122177`.

Sample Input 3

```
15 10
61477244 450343304 812961384 836482955 280670539 405068748 318805088 304825858 518212597 316347783 589272551 505875419 944071276 364842194 537
6942
2 11
5 9
8 15
6 7
6 8
1 2
1 10
4 9
12 15
6 11
```

Sample Output 3

```
449356308 449356308 449356308 449356308 449356308 648148154 648148154 648148154 648148154 648148154 648148154 643863031 643863031 643863031 64
3863031
```

G - Binary Cat

Time Limit: 6 sec / Memory Limit: 1024 MiB

Score : 625 points

Problem Statement

Define strings S_0 and S_1 as $S_0 =$ `0` and $S_1 =$ `1`.

You are given Q queries, so process them in order.
In the i-th query $(1 \le i \le Q)$, you are given a triple of integers (L_i, R_i, X_i).

> Let S_{i+1} be the string obtained by concatenating S_{L_i} and S_{R_i} in this order. Then, find the X_i-th character of S_{i+1}.
>
> It is guaranteed that X_i is at most the length of S_{i+1}.

Constraints

- $1 \le Q \le 5 \times 10^5$
- $0 \le L_i, R_i \le i$
- $1 \le X_i \le 10^{18}$
- X_i is at most the length of S_{i+1}.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
Q \\
L_1 \quad R_1 \quad X_1 \\
L_2 \quad R_2 \quad X_2 \\
\vdots \\
L_Q \quad R_Q \quad X_Q
\end{array}
$$

Output

Output Q lines. The i-th line $(1 \le i \le Q)$ should contain the answer to the i-th query.

Sample Input 1

```
7
0 1 1
0 0 2
1 1 1
2 3 2
2 4 3
5 4 2
6 7 6
```

Sample Output 1

```
0
0
1
1
1
1
1
```

Each query is processed as follows:

- In the first query, concatenate S_0 $=$`0` and S_1 $=$`1` to get S_2 $=$`01`. The first character of S_2 is `0`, so output 0 on the first line.
- In the second query, concatenate S_0 $=$`0` and S_0 $=$`0` to get S_3 $=$`00`. The second character of S_3 is `0`, so output 0 on the second line.
- In the third query, concatenate S_1 $=$`1` and S_1 $=$`1` to get S_4 $=$`11`. The first character of S_4 is `1`, so output 1 on the third line.
- In the fourth query, concatenate S_2 $=$`01` and S_3 $=$`00` to get S_5 $=$`0100`. The second character of S_5 is `1`, so output 1 on the fourth line.
- In the fifth query, concatenate S_2 $=$`01` and S_4 $=$`11` to get S_6 $=$`0111`. The third character of S_6 is `1`, so output 1 on the fifth line.
- In the sixth query, concatenate S_5 $=$`0100` and S_4 $=$`11` to get S_7 $=$`010011`. The second character of S_7 is `1`, so output 1 on the sixth line.
- In the seventh query, concatenate S_6 $=$`0111` and S_7 $=$`010011` to get S_8 $=$`0111010011`. The sixth character of S_8 is `1`, so output 1 on the seventh line.