

D. Team Arrangement

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Recently personal training sessions have finished in the Berland State University Olympiad Programmer Training Centre. By the results of these training sessions teams are composed for the oncoming team contest season. Each team consists of three people. All the students of the Centre possess numbers from 1 to $3n$, and all the teams possess numbers from 1 to n . The splitting of students into teams is performed in the following manner: while there are people who are not part of a team, a person with the best total score is chosen among them (the captain of a new team), this person chooses for himself two teammates from those who is left according to his list of priorities. The list of every person's priorities is represented as a permutation from **the rest of** $3n - 1$ students who attend the centre, besides himself.

You are given the results of personal training sessions which are a permutation of numbers from 1 to $3n$, where the i -th number is the number of student who has won the i -th place. No two students share a place. You are also given the arrangement of the already formed teams in the order in which they has been created. Your task is to determine the list of priorities for the student number k . If there are several priority lists, choose the lexicographically minimal one.

Input

The first line contains an integer n ($1 \leq n \leq 10^5$) which is the number of resulting teams. The second line contains $3n$ space-separated integers from 1 to $3n$ which are the results of personal training sessions. It is guaranteed that every student appears in the results exactly once.

Then follow n lines each containing three integers from 1 to $3n$ — each line describes the members of a given team. The members of one team can be listed in any order, but the teams themselves are listed in the order in which they were created. It is guaranteed that the arrangement is correct, that is that every student is a member of exactly one team and those teams could really be created from the given results using the method described above.

The last line contains number k ($1 \leq k \leq 3n$) which is the number of a student for who the list of priorities should be found.

Output

Print $3n - 1$ numbers — the lexicographically smallest list of priorities for the student number k .

The lexicographical comparison is performed by the standard $<$ operator in modern programming languages. The list a is lexicographically less than the list b if exists such an i ($1 \leq i \leq 3n$), that $a_i < b_i$, and for any j ($1 \leq j < i$) $a_j = b_j$. Note, that the list 1 9 10 is lexicographically less than the list 1 10 9. That is, the comparison of lists is different from the comparison of lines.

Examples

input
3 5 4 1 2 6 3 7 8 9 5 6 2 9 3 4 1 7 8 4
output
2 3 5 6 9 1 7 8

input

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

output

1 2 3 4 5 6 7 9

input

2
4 1 3 2 5 6
4 6 5
1 2 3
4

output

5 6 1 2 3