

2024 KAIST RUN Spring Contest

Sponsored By:



Rules

- This contest is 2024 KAIST RUN Spring Contest.
- This contest is sponsored by Jane Street, Startlink, and Samsung Software Membership.
- You can only participate individually.
- Use of the network is prohibited during the competition, except for submitting source codes and accessing language reference sites, and using translation. Here are examples of allowed reference and translation sites:
 - C/C++ : <https://en.cppreference.com/w/>
 - Java : <https://docs.oracle.com/javase/8/docs/api/>
 - Python : <https://docs.python.org/>
 - Kotlin : <https://kotlinlang.org/docs/reference/>
 - Naver Papago : <https://papago.naver.com/>
 - Google Translate: <https://translate.google.com/>
- Use of pre-written code or template, any library is prohibited. You should only submit code written by yourself during contest time.
- The contest lasts 4 hours.
- The contest consists of 8 problems.
- Problems are ordered roughly by the difficulty of getting full points. You are strongly encouraged to read all the problems.
- Each problem consists of 1 or more subtasks, and each subtask is worth a certain number of points. Subtasks are easier versions of the problems with more restrictive constraints.
- Each subtask consists of several tests, and points for the subtask are awarded if every test is passed. The total number of points for each problem is 100 points.
- Each problem has time limit and memory limit. This means your problem should run in the given time and memory limit for each test.
- Every problem is guaranteed to be solvable using C++17. This is not guaranteed for any other language.
- The memory limit for every problem is 1024 MiB.
- Each participant's ranking is determined in the following way:
 - Penalty time = (Duration from contest start to the last submission that increased points)
 - Ranking = (# of participants with higher points) + (# of participants with same points and lower penalty time) + 1

Problem list

#	Problem name	Time limit (All languages)
A	RUN Number	1 second
B	Sequence and Queries	1 second
C	Construct a Graph	1 second
D	Closet	5 seconds
E	Two Histograms	2 seconds
F	Discount Event	4 seconds
G	One, Two, Three	1 second
H	Tree Kadane	3 seconds

Problem A. RUN Number

Time limit: 1 second

양의 정수 x 의 모든 자릿수가 같은 숫자로 이루어져 있다면, 그런 x 를 **RUN** 수라고 한다. 예를 들어, 4, 111, 888 888은 **RUN** 수이지만, 27, 334, 100 000은 아니다.

N 자리 수 K 가 주어지면, K 를 최대 $(N + 1)$ 개의 **RUN** 수의 합으로 표현하라. 이는 항상 가능함을 증명할 수 있다.

Input

첫 번째 줄에는 테스트 케이스의 개수 T 가 주어진다.

각 테스트 케이스는 두 정수 N 과 K 가 한 줄에 공백으로 구분되어 주어진다.

Output

각 테스트 케이스에 대해서,

- 첫 번째 줄에는 더해서 K 를 만들 **RUN** 수의 개수 M 을 출력한다. ($1 \leq M \leq N + 1$)
- 두 번째 줄에는 해당하는 M 개의 **RUN** 수를 공백으로 구분해서 출력한다. 만약 답이 여러 개 존재한다면, 그 중 아무거나 출력하면 된다.

Constraints

- $1 \leq T \leq 100$
- $1 \leq N \leq 17$
- $10^{N-1} \leq K < 10^N$. 즉, K 는 N 자리 수이다.

Subtasks

- 서브태스크 1 (5점): $N \leq 2$
- 서브태스크 2 (10점): $N \leq 5$
- 서브태스크 3 (25점): $K = \overline{s_1 s_2 \dots s_N}$ 로 표현하면, $s_1 \leq s_2 \leq \dots \leq s_{N-1} \leq s_N$ 을 만족한다. 즉, K 의 각 자리 숫자는 단조증가한다.
- 서브태스크 4 (60점): 추가적인 제약 조건이 없다.

Examples

standard input	standard output
2 4 2024 3 506	4 999 999 22 4 3 444 55 7
2 17 99999999999999999 9 333666999	1 99999999999999999 3 333333333 333333 333

Problem B. Sequence and Queries

Time limit: 1 second

길이가 n 인 수열 (s_1, s_2, \dots, s_n) 이 주어진다. 함수 f 는 다음과 같이 정의된다.

$$f(i, j, k) = \begin{cases} 1 & \text{if } s_{i+t} \leq s_{j+t} \text{ for all } 0 \leq t < k \\ 0 & \text{otherwise} \end{cases}$$

$\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^{\min(n-i+1, n-j+1)} f(i, j, k)$ 의 값을 출력하라.

Input

첫 번째 줄에 수열의 길이를 나타내는 정수 n 이 주어진다.

두 번째 줄에 n 개의 정수 s_1, s_2, \dots, s_n 이 공백으로 구분되어 주어진다.

Output

$\sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^{\min(n-i+1, n-j+1)} f(i, j, k)$ 의 값을 출력한다.

Constraints

- $1 \leq n \leq 5\,000$
- $1 \leq s_x \leq 10^9$ ($1 \leq x \leq n$)

Subtasks

- 서브태스크 1 (8점): $1 \leq n \leq 500$
- 서브태스크 2 (92점): 추가적인 제약 조건이 없다.

Examples

standard input	standard output
2 2 3	4
5 2 4 2 2 1	35

Note

첫 번째 예제에 대한 설명은 다음과 같다.

- $f(1, 1, 1) = 1$
- $f(1, 1, 2) = 1$
- $f(1, 2, 1) = 1$
- $f(2, 1, 1) = 0$
- $f(2, 2, 1) = 1$

Problem C. Construct a Graph

Time limit: 1 second

$N \times N$ 크기의 행렬 D 가 있다. 당신은 정점이 N 개이고 아래 조건들을 만족하는 무방향 연결 그래프를 구성해야 한다. 각 정점은 1부터 N 까지 번호가 매겨져 있으며, 각 간선에는 양의 정수 가중치를 원하는 대로 부여할 수 있다.

- 모든 정점 쌍 (u, v) 에 대해, u 와 v 사이의 최단 경로의 길이는 $D_{u,v}$ 이다.
- 모든 간선의 가중치의 합은 가능한 최소여야 한다.

조건을 만족하는 그래프가 존재하는지 판별하고, 있다면 그 중 아무거나 하나를 출력하라.

Input

첫 번째 줄에 정점의 개수를 나타내는 정수 N 이 주어진다.

다음 N 개 줄 중 i 번째 줄에는 N 개의 정수 $D_{i,1}, D_{i,2}, \dots, D_{i,N}$ 이 공백으로 구분되어 주어진다.

Output

문제의 조건을 만족하는 그래프가 존재하지 않는다면, -1 을 출력한다.

조건을 만족하는 그래프가 존재한다면,

- 첫 번째 줄에 간선의 개수를 나타내는 정수 M 을 출력한다.
- 다음 M 개 줄 중 i 번째 줄에 세 정수 u_i, v_i, c_i 를 공백으로 구분하여 출력한다. 이것은 i 번째 간선이 두 정점 u_i 와 v_i 를 잇고 가중치가 c_i 라는 것을 나타낸다.
- 같은 쌍의 정점을 연결하는 간선은 최대 하나여야 하고, 각 간선의 가중치는 10^9 이하여야 한다.

Constraints

- $2 \leq N \leq 300$
- $D_{i,i} = 0$ ($1 \leq i \leq N$)
- $1 \leq D_{i,j} = D_{j,i} \leq 10^9$ ($1 \leq i < j \leq N$)
- $1 \leq u_i, v_i \leq N$ ($1 \leq i \leq M$)
- $u_i \neq v_i$ ($1 \leq i \leq M$)
- $1 \leq c_i \leq 10^9$ ($1 \leq i \leq M$)
- 같은 쌍의 정점을 연결하는 간선은 최대 하나여야 한다.

Subtasks

- 서브태스크 1 (9점): 이 서브태스크에서는 조건을 만족하는 그래프가 존재하지만 판별해도 된다. 즉, 채점 프로그램은 출력이 -1 인지 아닌지의 여부만 확인한다.
- 서브태스크 2 (19점): $N \leq 50$
- 서브태스크 3 (72점): 추가적인 제약 조건이 없다.

Examples

standard input	standard output
3 0 1 2 1 0 3 2 3 0	2 1 2 1 1 3 2
3 0 1 3 1 0 1 3 1 0	-1

Problem D. Closet

Time limit: 5 seconds

태인이의 옷장에는 N 개의 옷이 일렬로 걸려 있다. 현재 왼쪽에서 i 번째 옷의 색은 c_i 이다.

태인이는 어떤 정수 k ($1 \leq k \leq N$)가 존재해 $c_1 \leq c_2 \leq \dots \leq c_k \geq c_{k+1} \geq \dots \geq c_N$ 를 만족하면 옷장이 아름답다고 생각한다.

하지만 옷장을 아름다운 상태로 정리하는 것은 꽤 귀찮다. 그래서 태인이는 옷장에서 최대 M 개의 옷을 제거해 남은 옷들을 거의 아름다운 상태로 만들기로 했다.

최대 M 개의 옷을 제거한 후, 남은 옷들의 개수를 L , 왼쪽에서 j 번째 옷의 색을 d_j 라 하자. 태인이는 인접한 두 옷의 색의 차가 x 이하라면 두 옷을 같은 색으로 인식하기로 했다. 즉, 다음을 만족하는 정수 k ($1 \leq k \leq L$)가 존재한다면 옷장이 거의 아름다운 상태라고 한다.

- $1 \leq j < k$ 인 모든 정수 j 에 대해 $d_j - d_{j+1} \leq x$.
- $k \leq j < L$ 인 모든 정수 j 에 대해 $d_{j+1} - d_j \leq x$.

M 개 이하의 옷을 제거해 옷장을 거의 아름다운 상태로 만들 수 있는 가장 작은 음이 아닌 정수 x 의 값을 구하자.

Input

첫 번째 줄에 두 정수 N 과 M 이 주어진다.

두 번째 줄에 N 개의 정수 c_1, c_2, \dots, c_N 가 공백을 사이에 두고 주어진다.

Output

옷장을 거의 아름다운 상태로 만들 수 있는 가장 작은 x ($x \geq 0$) 의 값을 출력한다.

Constraints

- $1 \leq N \leq 10^5$
- $0 \leq M \leq N$
- $1 \leq c_i \leq 10^9$ ($1 \leq i \leq N$)

Subtasks

- 서브태스크 1 (4점): $M = 0$
- 서브태스크 2 (21점): $1 \leq N \leq 1000$
- 서브태스크 3 (16점): $1 \leq c_i \leq 100$ ($1 \leq i \leq N$)
- 서브태스크 4 (59점): 추가적인 제약 조건이 없다.

Example

standard input	standard output
10 2 4 2 7 15 3 11 12 10 2 6	2

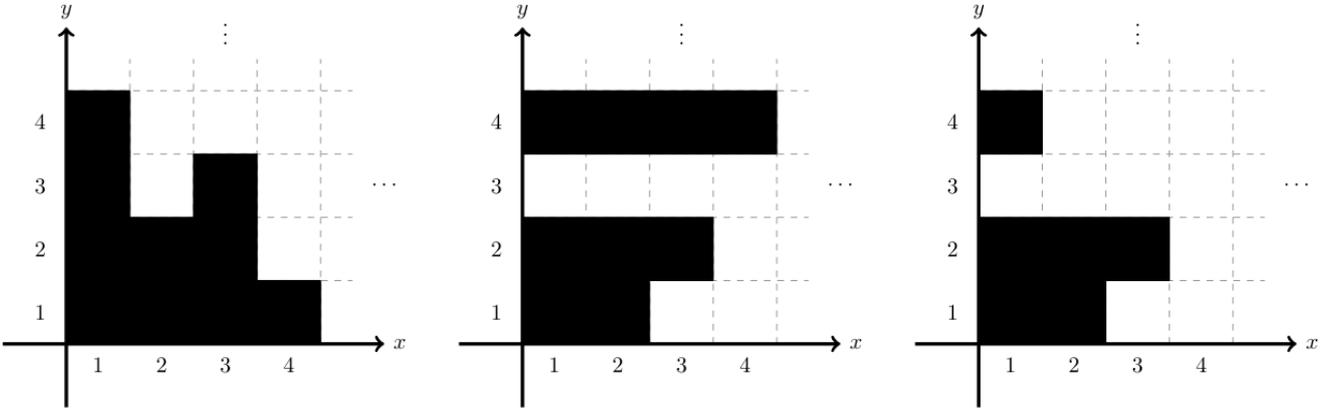
Note

$x = 2$ 일 때, 왼쪽에서 5 번째 옷과 9 번째 옷을 제거하면 옷장은 거의 아름다운 상태가 된다. 이보다 더 작은 x 로는 옷장을 거의 아름다운 상태로 만들 수 없다.

Problem E. Two Histograms

Time limit: 2 seconds

당신에게 $10^6 \times 10^6$ 크기의 정사각형 모양의 격자판 세 개가 주어진다. 각 칸은 x 좌표와 y 좌표로 번호가 매겨져 있다. x 좌표는 맨 왼쪽에서부터 맨 오른쪽까지 1부터 10^6 으로 매겨져 있고, y 좌표는 맨 아래에서부터 맨 위까지 1에서 10^6 으로 매겨져 있다. 당신은 각 칸을 검은색 혹은 흰색으로 칠해야 한다.



세 격자의 격자칸을 색칠하는 예시.

첫 번째 격자판은 아래에서부터 올라오는 히스토그램의 형태를 띄어야 한다. 즉, 어떤 격자칸이 검은색으로 칠해져 있다면, 그 아래의 칸도 검은색으로 칠해져 있어야 한다.

두 번째 격자판은 왼쪽에서부터 오른쪽으로 진행되는 히스토그램의 형태를 띄어야 한다. 즉, 어떤 격자칸이 검은색으로 칠해져 있다면, 그 왼쪽 칸도 검은색으로 칠해져 있어야 한다.

세 번째 격자판은 앞의 두 격자판을 이용해 색칠한다. 어떤 칸 (x, y) 가 첫 두 격자판에서 모두 검은색으로 색칠되어 있다면, 세 번째 격자판의 칸 (x, y) 역시 검은색으로 색칠한다. 그렇지 않다면, 해당 칸을 흰색으로 색칠한다. 이 세 번째 격자판이 최종 그림이 된다.

당신이 그린 그림을 N 명이 심사위원에게 심사할 예정이다. 각 심사위원은 그림 내의 특정한 $K \times 1$ 직사각형 영역을 심사에 이용한다. i 번째 심사위원이 이용하는 직사각형 영역은 $[x_i, x_i + K - 1] \times [y_i, y_i]$ 이다. 각 심사위원들이 심사에 이용하는 직사각형 영역은 겹치지 않는다.

i 번째 심사위원은 칸 (x_i, y_i) 와 칸 $(x_i + K - 1, y_i)$ 가 같은 색으로 칠해진 경우 불합격으로 판정한다. 두 칸의 색이 다른 경우에는 합격으로 판정하고, 칸 (x_i, y_i) 가 흰색으로 칠해진 경우에 a_i 점을, 검은색으로 칠해진 경우에 b_i 점을 준다.

심사를 통과하기 위해서는 모든 심사위원에게 합격 판정을 받아야 한다. 이때 그림의 점수는 모든 심사위원들에게 받은 점수의 합이 된다. 심사를 통과하는 가능한 모든 그림에 대해서 받을 수 있는 점수의 최댓값을 구해 보자.

Input

첫 번째 줄에는 두 정수 N 과 K 가 공백으로 구분되어 주어진다.

다음 N 개의 줄 중 i 번째 줄에는 네 정수 x_i, y_i, a_i, b_i 가 공백으로 구분되어 주어진다.

Output

심사를 통과하는 그림이 없다면, -1 을 출력한다.

심사를 통과하는 그림이 있다면, 가능한 그림의 최대 점수를 출력한다.

Constraints

- $1 \leq N \leq 3 \times 10^5$
- $2 \leq K \leq 10^6$
- $1 \leq x_i \leq 10^6 - K + 1$ ($1 \leq i \leq N$)

- $1 \leq y_i \leq 10^6$ ($1 \leq i \leq N$)
- $1 \leq a_i, b_i \leq 10^9$ ($1 \leq i \leq N$)
- N 개의 직사각형 영역 $[x_i, x_i + K - 1] \times [y_i, y_i]$ ($1 \leq i \leq N$) 는 서로 겹치지 않는다.

Subtasks

- 서브태스크 1 (30점): $K = 2; N \leq 5000$
- 서브태스크 2 (30점): $K = 2$
- 서브태스크 3 (40점): 추가적인 제약 조건이 없다.

Examples

standard input	standard output
5 2 1 1 1 2 3 1 1 10 5 1 5 6 2 2 3 2 4 2 5 9	26
6 3 1 1 2 4 5 1 4 9 2 3 7 4 5 3 3 1 1 5 5 7 4 5 6 4	36
10 2 7 2 2 4 4 4 6 3 1 5 1 4 3 5 2 8 5 2 4 3 6 4 4 2 1 2 1 4 5 6 9 7 7 1 6 3 4 3 8 7	51
10 3 4 2 5 2 10 2 8 10 1 2 1 4 12 1 8 6 6 3 7 10 8 1 1 9 11 3 5 5 7 2 10 5 3 3 6 4 4 1 9 4	72

Note

$[x_l, x_r] \times [y_l, y_r]$ 직사각형 영역은 $x_l \leq x \leq x_r$ 이고 $y_l \leq y \leq y_r$ 인 영역을 의미한다.

Problem F. Discount Event

Time limit: 4 seconds

N 개의 도시와 $N - 1$ 개의 도로로 이루어진 국가가 있다. 도시에는 1 부터 N 까지의 번호가 붙어 있고, 도로에도 1 부터 $N - 1$ 의 번호가 붙어 있다. i 번 도로는 A_i 번 도시와 B_i 번 도시를 양방향으로 연결하고, 이동 시에는 W_i 의 비용이 든다. 임의의 두 도시를 고르더라도 둘 사이를 하나 이상의 도로를 사용하여 왕복할 수 있음이 보장된다.

두 도시 사이의 거리를 한 도시에서 출발하여 하나 이상의 도로를 거쳐 다른 도시로 갈 때 필요한 최소 비용으로 정의하자.

당신은 도로 회사의 사장으로, 명절을 맞아 할인 행사를 진행하려고 한다. 할인 행사를 위한 총 Q 개의 계획이 있다. i 번째 계획에서는 X_i 번 도시에서 출발하여 Y_i 번 도시로 가는 최단 경로에 속하는 모든 도로들에 할인을 적용하여 비용을 0으로 만들 것이다. 각 할인 행사 계획에 대해, 두 도시 사이 거리의 최댓값을 출력하라.

Input

첫 번째 줄에 도시의 수를 나타내는 정수 N 이 주어진다.

다음 $N - 1$ 개의 줄 중 i 번째 줄에는 세 정수 A_i, B_i, W_i 가 공백으로 구분되어 주어진다. 이들은 i 번째 도로가 연결하는 두 도시의 번호와 도로의 이동 비용을 나타낸다.

다음 줄에 계획의 수를 나타내는 정수 Q 가 주어진다.

다음 Q 개의 줄 중 i 번째 줄에는 두 정수 X_i 와 Y_i 가 공백으로 구분되어 주어진다. 이들은 i 번째 계획을 나타낸다.

Output

총 Q 개의 줄에 걸쳐 답을 출력한다. i 번째 줄에는 i 번째 계획에 대한 답을 출력해야 한다.

Constraints

- $2 \leq N \leq 100\,000$
- $1 \leq A_i, B_i \leq N$ ($1 \leq i \leq N - 1$)
- $A_i \neq B_i$ ($1 \leq i \leq N - 1$)
- $1 \leq W_i \leq 10^9$ ($1 \leq i \leq N - 1$)
- 입력으로 주어지는 나라의 구조는 올바른 트리를 이룬다.
- $1 \leq Q \leq 100\,000$
- $1 \leq X_i, Y_i \leq N$ ($1 \leq i \leq Q$)
- $X_i \neq Y_i$ ($1 \leq i \leq Q$)

Subtasks

- 서브태스크 1 (10 점): $N, Q \leq 2\,000$
- 서브태스크 2 (24 점): $X_i = 1$ ($1 \leq i \leq Q$)
- 서브태스크 3 (27 점): $Q = N - 1$; $X_i = A_i$; $Y_i = B_i$ ($1 \leq i \leq N - 1$)
- 서브태스크 4 (39 점): 추가적인 제약 조건이 없다.

Examples

standard input	standard output
5 1 2 1 1 3 2 2 4 4 2 5 2 3 1 4 3 5 5 4	4 4 3
5 1 2 2 1 3 3 1 4 5 1 5 4 3 1 2 1 4 4 5	9 7 5
8 1 2 3 1 3 5 2 4 5 2 5 3 2 6 6 6 7 1 6 8 2 7 1 2 1 3 2 4 2 5 2 6 6 7 6 8	13 13 16 16 13 16 15
7 1 2 3 1 3 4 2 4 2 2 5 1 3 6 6 3 7 2 6 1 2 1 3 1 4 1 5 1 6 1 7	12 11 11 12 7 11

Problem G. One, Two, Three

Time limit: 1 second

길이 N 인 수열 a_1, a_2, \dots, a_N 이 주어진다. a 의 각 원소는 1, 2, 또는 3이다.

각 원소는 양의 정수로 나타나는 아름다움을 가지고 있다. i 번째 원소의 아름다움은 b_i 로 나타난다.

선택한 구간 안에 있는 원소의 합이 4 혹은 8이면 구간 내 원소들을 제거하는 시행을 원하는 만큼 할 수 있다.

남아있는 원소들의 합을 최소화하려고 한다. 만약 원소들의 합을 최소화시키는 방법이 여러 가지 있다면, 남아있는 원소들의 아름다움의 합을 최대화하려고 한다.

수열에 원하는 만큼 시행을 하여 남아있는 원소들의 합을 최소화해보자. 만약 남아있는 원소들의 합을 최소화시키는 방법이 여러 가지 있다면, 남아있는 원소들의 아름다움의 합을 최대화하자. 모든 테스트 케이스에 대해서 실행한 시행의 목록을 출력하지 않아도 됨에 유의하라.

Input

각 입력은 여러 개의 테스트 케이스로 이루어져 있다. 첫 번째 줄에는 테스트 케이스의 개수 T 가 주어진다.

각 테스트 케이스는 4개의 줄로 이루어져 있다.

- 첫 번째 줄에는 수열의 길이를 나타내는 정수 N 이 주어진다.
- 두 번째 줄에는 처음 수열을 나타내는 문자열 $a_1a_2\dots a_N$ 이 주어진다. 각 문자는 1, 2, 3 중 하나이다.
- 세 번째 줄에는 각 원소의 아름다움을 나타내는 N 개의 정수 b_1, b_2, \dots, b_N 이 공백으로 구분되어 주어진다.
- 네 번째 줄에는 테스트 케이스의 유형을 나타내는 정수 p 가 주어진다. $p = 1$ 인 테스트 케이스의 경우, 남아있는 원소들의 합의 최솟값과 남아있는 원소들의 아름다움의 합을 최댓값만 출력하면 된다. $p = 2$ 인 경우에는 실행한 시행들의 목록도 출력해야 한다.

Output

각 테스트 케이스에 대해서, 첫 번째 줄에 남아있는 원소들의 합의 최솟값과 남아있는 원소들의 아름다움의 합의 최댓값을 출력한다.

$p = 2$ 인 경우에, 실행한 시행들의 목록도 출력해야 한다. 두 번째 줄에 실행한 시행의 개수 K 를 출력한다.

다음에 이어지는 K 개의 줄에는 각 시행에 대해 출력해야 한다. i 번째 줄에는 i 번째 시행에 제거한 원소의 개수를 출력한다. 그리고 제거한 원소들의 위치를 오름차순으로 출력한다. 각 시행마다 제거한 원소들은 i 번째 시행을 실행하기 전에 수열에서 구간을 이루어야 한다. 제거한 원소들의 위치를 출력할 때 처음에 주어진 수열에서의 위치를 출력해야 한다.

Constraints

- $1 \leq T \leq 2000$
- 각 테스트 케이스에 대해 $N \geq 1$
- $\sum N \leq 3 \times 10^5$
- 각 테스트 케이스에 대해 $1 \leq a_i \leq 3$ ($1 \leq i \leq N$)
- 각 테스트 케이스에 대해 $1 \leq b_i \leq 99$ ($1 \leq i \leq N$)
- 각 테스트 케이스에 대해 $1 \leq p \leq 2$

Subtasks

- 서브태스크 1 (20점): $\sum N \leq 3000$. 각 테스트 케이스에 대해 $p = 1$.
- 서브태스크 2 (30점): $\sum N \leq 3000$
- 서브태스크 3 (20점): 각 테스트 케이스에 대해 $p = 1$
- 서브태스크 4 (30점): 추가적인 제약 조건이 없다.

Example

standard input	standard output
6	0 0
3	1
121	3 1 2 3
1 2 3	1 6
2	1
3	2 1 2
221	6 24
4 5 6	3 29
2	1
3	4 1 2 3 4
123	5 25
7 8 9	2
1	2 5 6
6	2 1 2
123221	0 0
10 11 12 13 14 15	3
2	2 2 3
6	2 4 5
312322	2 1 6
15 14 13 12 11 10	
2	
6	
213312	
9 8 7 5 5 1	
2	

Problem H. Tree Kadane

Time limit: 3 seconds

1부터 N 까지 번호가 매겨진 N 개의 정점으로 이루어진 트리 T 가 주어진다. 각 i 번 정점은 정수 가중치 A_i 를 가진다.

T 의 연결 부분 집합 S 는 T 의 정점들의 공집합이 아닌 부분 집합으로, S 에 속하는 두 정점 a, b 에 대해 S 에 속한 정점만을 사용하는 a 에서 b 로 가는 경로가 T 에 존재한다.

다음 Q 개의 쿼리를 처리해야 한다.

- $k x$: A_k 를 x 로 바꾸고 T 의 연결 부분 집합 S 에 대한 $\sum_{i \in S} A_i$ 의 최댓값을 출력한다.

Input

첫 번째 줄에 정점의 개수를 나타내는 정수 N 이 주어진다.

두 번째 줄에 각 정점의 가중치를 나타내는 N 개의 정수 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다.

다음 $N - 1$ 개의 줄 중 i 번째 줄에 두 정수 u_i 와 v_i 가 공백으로 구분되어 주어진다. 이들은 i 번째 간선의 양 끝 점을 나타낸다.

다음 줄에 쿼리의 개수를 나타내는 정수 Q 가 주어진다.

다음 Q 개의 줄에 걸쳐 각 줄에 각 쿼리를 나타내는 두 정수 k 와 x 가 공백으로 구분되어 주어진다.

Output

Q 개의 줄에 걸쳐 쿼리의 정답을 한 줄에 하나씩 출력한다.

Constraints

- $1 \leq N \leq 10^5$
- $-10^9 \leq A_i \leq 10^9$ ($1 \leq i \leq N$)
- $1 \leq u_i, v_i \leq N$ ($1 \leq i \leq N - 1$)
- 주어지는 그래프는 트리이다.
- $1 \leq Q \leq 10^5$
- 각 쿼리에 대해 $1 \leq k \leq N$
- 각 쿼리에 대해 $-10^9 \leq x \leq 10^9$

Subtasks

- 서브태스크 1 (10점): $N, Q \leq 10^3$
- 서브태스크 2 (20점): 각 정점의 차수는 3 미만이다.
- 서브태스크 3 (70점): 추가적인 제약 조건이 없다.

Example

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