**문제**

연인 코니와 브라운은 광활한 들판에서 ‘나 잡아 봐라’ 게임을 한다. 이 게임은 브라운이 코니를 잡거나, 코니가 너무 멀리 달아나면 끝난다. 게임이 끝나는데 걸리는 최소 시간을 구하시오.

**조건**

1. 코니는 처음 위치 C에서 1초 후 1만큼 움직이고, 이후에는 가속이 붙어 매 초마다 이전 이동 거리 + 1만큼 움직인다. 즉 시간에 따른 코니의 위치는 C, C + 1, C + 3, C + 6, …이다.
2. 브라운은 현재 위치 B에서 다음 순간 B – 1, B + 1, 2 \* B 중 하나로 움직일 수 있다.
3. 코니와 브라운의 위치 p는 조건 0 <= x <= 200,000을 만족한다.
4. 브라운은 범위를 벗어나는 위치로는 이동할 수 없고, 코니가 범위를 벗어나면 게임이 끝난다.

**입력 형식**

표준 입력의 첫 줄에 코니의 위치 C와 브라운의 위치 B를 공백으로 구분하여 순서대로 읽는다.

**출력 형식**

브라운이 코니를 잡을 수 있는 최소시간 N초를 표준 출력한다. 단 브라운이 코니를 잡지 못한 경우에는 -1을 출력한다.

**예제**

입력: 11 2

출력: 5

코니의 위치: 11 → 12 → 14 → 17 → 21 → 26

브라운의 위치: 2 → 3 → 6 → 12 → 13 → 26

브라운은 코니를 5초 만에 잡을 수 있다.

트리와 쿼리

## **문제**

간선에 가중치와 방향성이 없는 임의의 루트 있는 트리가 주어졌을 때, 아래의 쿼리에 답해보도록 하자.

* 정점 U를 루트로 하는 서브트리에 속한 정점의 수를 출력한다.

만약 이 문제를 해결하는 데에 어려움이 있다면, 하단의 힌트에 첨부한 문서를 참고하자.

## **입력**

트리의 정점의 수 N과 루트의 번호 R, 쿼리의 수 Q가 주어진다. (2 ≤ N ≤ 105, 1 ≤ R ≤ N, 1 ≤ Q ≤ 105)

이어 N-1줄에 걸쳐, U V의 형태로 트리에 속한 간선의 정보가 주어진다. (1 ≤ U, V ≤ N, U ≠ V)

이는 U와 V를 양 끝점으로 하는 간선이 트리에 속함을 의미한다.

이어 Q줄에 걸쳐, 문제에 설명한 U가 하나씩 주어진다. (1 ≤ U ≤ N)

입력으로 주어지는 트리는 항상 올바른 트리임이 보장된다.

## **출력**

Q줄에 걸쳐 각 쿼리의 답을 정수 하나로 출력한다.

## **예제 입력 1 복사**

9 5 3

1 3

4 3

5 4

5 6

6 7

2 3

9 6

6 8

5

4

8

## **예제 출력 1 복사**

9

4

1

## **힌트**

트리의 독립 집합

## **문제**

그래프 G(V, E)에서 정점의 부분 집합 S에 속한 모든 정점쌍이 서로 인접하지 않으면 (정점쌍을 잇는 에지가 없으면) S를 독립 집합(independent set)이라고 한다. 독립 집합의 크기는 정점에 가중치가 주어져 있지 않을 경우는 독립 집합에 속한 정점의 수를 말하고, 정점에 가중치가 주어져 있으면 독립 집합에 속한 정점의 가중치의 합으로 정의한다. 독립 집합이 공집합일 때 그 크기는 0이라고 하자. 크기가 최대인 독립 집합을 최대 독립 집합이라고 한다.

문제는 일반적인 그래프가 아니라 트리(연결되어 있고 사이클이 없는 그래프)와 각 정점의 가중치가 양의 정수로 주어져 있을 때, 최대 독립 집합을 구하는 것이다.

## **입력**

첫째 줄에 트리의 정점의 수 n이 주어진다. n은 10,000이하인 양의 정수이다. 1부터 n사이의 정수가 트리의 정점이라고 가정한다. 둘째 줄에는 n개의 정수 w1, w2, ..., wn이 주어지는데, wi는 정점 i의 가중치이다(1 ≤ i ≤ n). 셋째 줄부터 마지막 줄까지는 에지 리스트가 주어지는데, 한 줄에 하나의 에지를 나타낸다. 에지는 정점의 쌍으로 주어진다. 입력되는 정수들 사이에는 콤마가 없고 대신 빈칸이 하나 혹은 그 이상 있다. 가중치들의 값은 10,000을 넘지 않는 자연수이다.

## **출력**

첫째 줄에 최대 독립집합의 크기를 출력한다. 둘째 줄에는 최대 독립집합에 속하는 정점을 오름차순으로 출력한다. 최대 독립 집합이 하나 이상일 경우에는 하나만 출력하면 된다.

## **예제 입력 1 복사**

7

10 30 40 10 20 20 70

1 2

2 3

4 3

4 5

6 2

6 7

## **예제 출력 1 복사**

140

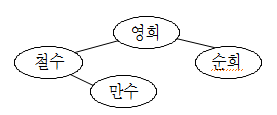
1 3 5 7

사회 관계망 서비스

## **문제**

페이스북, 트위터, 카카오톡과 같은 사회망 서비스(SNS)가 널리 사용됨에 따라, 사회망을 통하여 사람들이 어떻게 새로운 아이디어를 받아들이게 되는가를 이해하는 문제가 중요해졌다. 사회망에서 사람들의 친구 관계는 그래프로 표현할 수 있는데,  이 그래프에서 사람은 정점으로 표현되고, 두 정점을 잇는 에지는 두 정점으로 표현되는 두 사람이 서로 친구 관계임을 표현한다.

예를 들어, 철수와 영희, 철수와 만수, 영희와 순희가 서로 친구 관계라면 이를 표현하는 친구 관계 그래프는 다음과 같다.

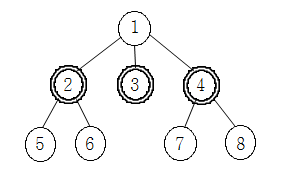


친구 관계 그래프를 이용하면 사회망 서비스에서 어떤 새로운 아이디어가 전파되는 과정을 이해하는데 도움을 줄 수 있다. 어떤 새로운 아이디어를 먼저 받아들인 사람을 얼리 아답터(early adaptor)라고 하는데, 사회망 서비스에 속한 사람들은 얼리 아답터이거나 얼리 아답터가 아니다. 얼리 아답터가 아닌 사람들은 자신의 모든 친구들이 얼리 아답터일 때만 이 아이디어를 받아들인다.

어떤 아이디어를 사회망 서비스에서 퍼뜨리고자 할 때, 가능한 한 최소의 수의 얼리 아답터를 확보하여 모든 사람이 이 아이디어를 받아들이게 하는  문제는 매우 중요하다.

일반적인 그래프에서 이 문제를 푸는 것이 매우 어렵다는 것이 알려져 있기 때문에, 친구 관계 그래프가 트리인 경우, 즉 모든 두 정점 사이에 이들을 잇는 경로가 존재하면서 사이클이 존재하지 않는 경우만 고려한다.

예를 들어, 8명의 사람으로 이루어진 다음 친구 관계 트리를 생각해보자. 2, 3, 4번 노드가 표현하는 사람들이 얼리 아답터라면, 얼리 아답터가 아닌 사람들은 자신의 모든 친구가 얼리 아답터이기 때문에 새로운 아이디어를 받아들인다.



친구 관계 트리가 주어졌을 때, 모든 개인이 새로운 아이디어를 수용하기 위하여 필요한 최소 얼리 어답터의 수를 구하는 프로그램을 작성하시오.

## **입력**

첫 번째 줄에는 친구 관계 트리의 정점 개수 N이 주어진다. 단, 2 <= N <= 1,000,000이며, 각 정점은 1부터 N까지 일련번호로 표현된다. 두 번째 줄부터 N-1개의 줄에는 각 줄마다 친구 관계 트리의 에지 (u, v)를 나타내는 두 정수 u 와 v가 하나의 빈칸을 사이에 두고 주어진다.

## **출력**

주어진 친구 관계 그래프에서 아이디어를 전파하는데 필요한 얼리 아답터의 최소 수를 하나의 정수로 출력한다.

## **예제 입력 1 복사**

8

1 2

1 3

1 4

2 5

2 6

4 7

4 8

## **예제 출력 1 복사**

3

우수마을

## **문제**

N개의 마을로 이루어진 나라가 있다. 편의상 마을에는 1부터 N까지 번호가 붙어 있다고 하자. 이 나라는 트리(Tree) 구조로 이루어져 있다. 즉 마을과 마을 사이를 직접 잇는 N-1개의 길이 있으며, 각 길은 방향성이 없어서 A번 마을에서 B번 마을로 갈 수 있다면 B번 마을에서 A번 마을로 갈 수 있다. 또, 모든 마을은 연결되어 있다. 두 마을 사이에 직접 잇는 길이 있을 때, 두 마을이 인접해 있다고 한다.

이 나라의 주민들에게 성취감을 높여 주기 위해, 다음 세 가지 조건을 만족하면서 N개의 마을 중 몇 개의 마을을 '우수 마을'로 선정하려고 한다.

1. '우수 마을'로 선정된 마을 주민 수의 총 합을 최대로 해야 한다.
2. 마을 사이의 충돌을 방지하기 위해서, 만일 두 마을이 인접해 있으면 두 마을을 모두 '우수 마을'로 선정할 수는 없다. 즉 '우수 마을'끼리는 서로 인접해 있을 수 없다.
3. 선정되지 못한 마을에 경각심을 불러일으키기 위해서, '우수 마을'로 선정되지 못한 마을은 적어도 하나의 '우수 마을'과는 인접해 있어야 한다.

각 마을 주민 수와 마을 사이의 길에 대한 정보가 주어졌을 때, 주어진 조건을 만족하도록 '우수 마을'을 선정하는 프로그램을 작성하시오.

## **입력**

첫째 줄에 정수 N이 주어진다. (1≤N≤10,000) 둘째 줄에는 마을 주민 수를 나타내는 N개의 자연수가 빈칸을 사이에 두고 주어진다. 1번 마을부터 N번 마을까지 순서대로 주어지며, 주민 수는 10,000 이하이다. 셋째 줄부터 N-1개 줄에 걸쳐서 인접한 두 마을의 번호가 빈칸을 사이에 두고 주어진다.

## **출력**

첫째 줄에 '우수 마을'의 주민 수의 총 합을 출력한다.

## **예제 입력 1 복사**

7

1000 3000 4000 1000 2000 2000 7000

1 2

2 3

4 3

4 5

6 2

6 7

## **예제 출력 1 복사**

14000