

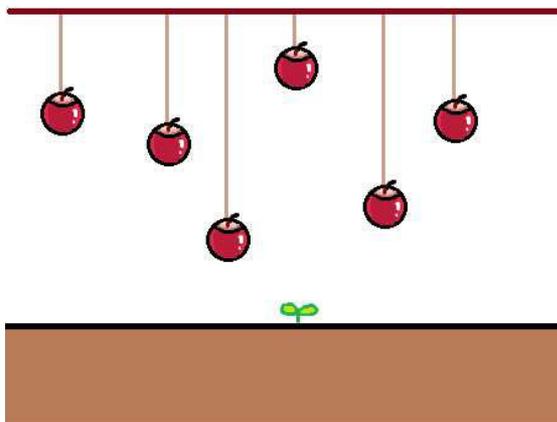
3. 2017 지구멸망

문제 유형 : Batch

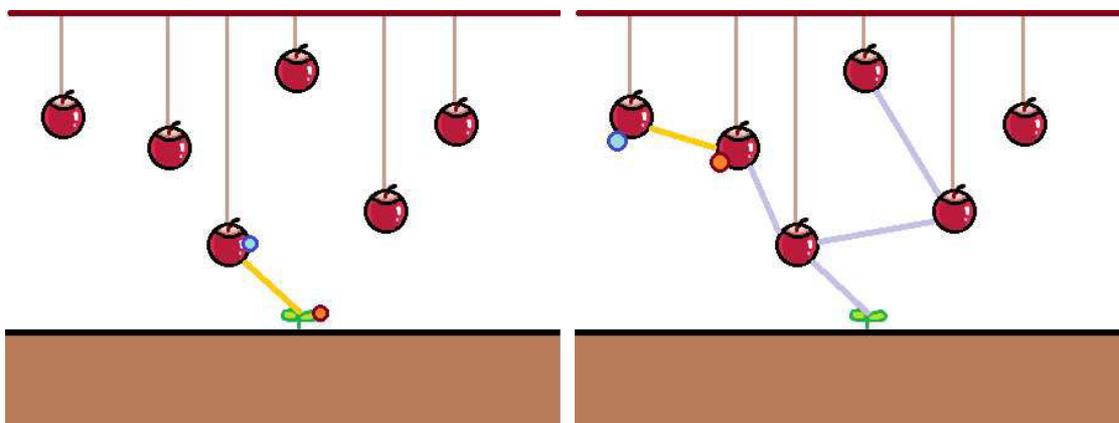
시간 제한 : 1 초, 메모리 제한 : 64 MB

2017년 7월 17일, 위대한 예언가 zigui의 예언에 따라 지구는 멸망할 것이다. 하지만 스피노자가 그랬던가, 내일 지구가 멸망하더라도 사과나무를 심을 거라고.

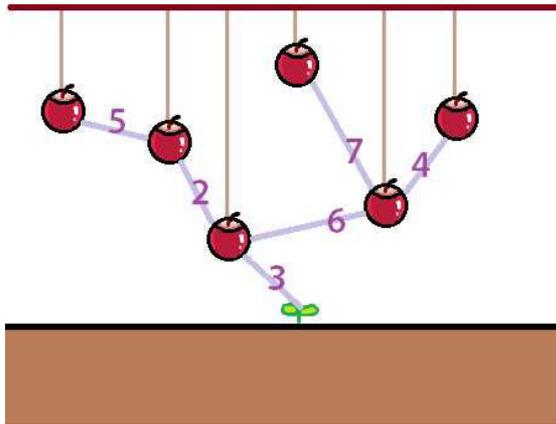
구사과는 지구멸망을 자축하기(?) 위해 지구멸망 하루 전인 7월 16일에 특별한 사과나무를 심을 것이다. 이 사과나무는 맨 처음에 아래 그림과 같이 땅 위에 새싹이 있으며 N개의 사과가 공중에 매달려 있다.



구사과는 특수 레이저 건을 이용해서 사과나무를 자라게 할 수 있다. 맨 처음에는 구사과가 새싹에 주황색 레이저를 쏘고, N개의 사과 중 하나에 하늘색 레이저를 쏜다. 그러면 새싹과 사과 사이에 줄기가 난다. 구사과가 레이저를 한 번 쏜 이후의 모습은 아래 그림에서 왼쪽 그림과 같다. 그 이후로도 구사과는 계속 줄기를 추가할 수 있다. 사과나무를 이루고 있는 사과 또는 새싹 부분에 주황색 레이저를 쏘고 사과나무를 이루고 있지 않은 사과에 하늘색 레이저를 쏘면 그 두 지점 사이에 줄기가 생긴다. 4개의 줄기가 생긴 사과나무에서 줄기를 하나 더 만드는 예는 아래 그림에서 오른쪽 그림과 같다. 만약 연결할 두 부분이 모두 사과나무를 이루고 있거나 모두 사과나무 이외이면 연결되지 않는다.

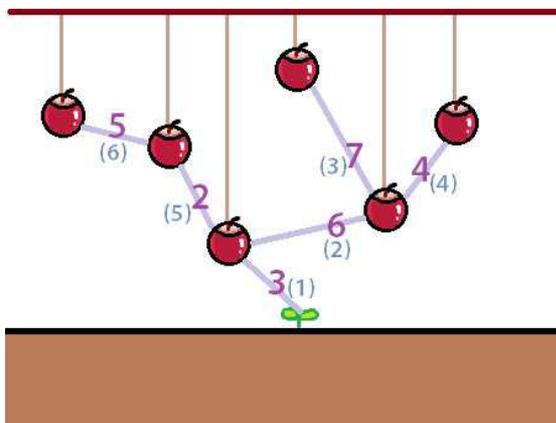


구사과는 아래 그림과 같이 자신이 원하는 사과나무의 모양을 하나 정해놓았다. 여기서 구사과가 정한 사과나무의 줄기의 수는 항상 N 개이다.

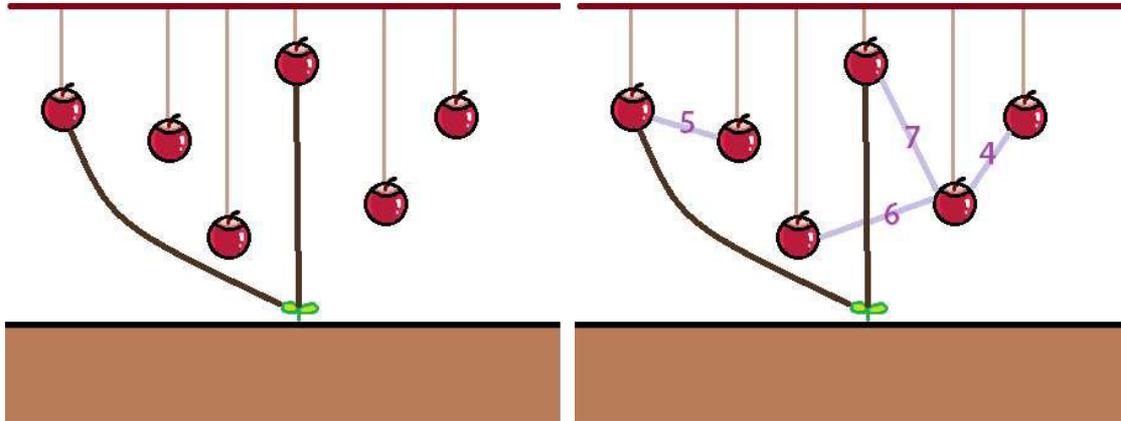


신기하게도, 구사과가 이런 방식으로 사과나무를 기르면 연보라색 줄기가 나타나서 반짝반짝 빛난다. 구사과는 각 줄기마다 예상 광도를 구해서 양의 정수로 나타냈는데, 재미있게도 줄기들의 광도는 모두 다르다.

구사과는 사과나무를 기를 때, 매 순간마다 자신이 만들 수 있는 줄기들 중 광도가 가장 높은 줄기를 선택하여 그 줄기를 기를 것이다. 구사과의 계획에 따르면, N 개의 줄기를 기르는 순서는 아래 그림에서 괄호로 표기되어 있다.



허나, 구사과가 깜빡하여 졸고 있는 동안 이미 사과나무가 자라 몇 개의 줄기가 생겼다. 이 줄기들은 모두 새싹과 사과를 연결하고 있으며 각 줄기가 연결하는 사과는 모두 다르다. 아래 그림에서 왼쪽 그림은 사과나무가 자란 형태로 가능한 예 중 하나이다. 왼쪽 그림의 사과나무에서 구사과가 가지를 더 자라게 하면 오른쪽 그림과 같은 형태가 된다.



사과나무에서 이미 자란 줄기들은 일반적인 갈색 줄기이므로, 빛이 나지 않는다. 따라서 사과가 다 자랐을 때 사과나무의 광도는 구사과가 추가로 기른 줄기들의 광도의 합이다. 또, 사과나무의 에너지는 사과나무의 광도의 제곱이다.

아직 구사과는 잠에서 깨지 않았다. 당신은 사과나무가 자랐을 수 있는 가능한 모든 형태에 대해, 구사과가 가지를 더 자라게 했을 때 만들어지는 최종 사과나무의 광도와 에너지를 구해 그 합을 구해야 한다. 그런 후 구사과를 깨워서 합을 알려주어 상황의 급박함을 깨닫게 해줄 것이다.

당신은 최대한 빠른 시간 안에 사과나무의 광도와 에너지의 합을 구해야 한다. 사과나무에서 줄기가 더 자라기 전에 어서 사과나무의 광도와 에너지의 합을 구하는 프로그램을 작성하자.

입력

첫 번째 줄에는 사과의 수 N 이 주어진다. ($1 \leq N \leq 300,000$)

두 번째 줄부터 N 개의 줄에는 구사과가 만들려고 하는 사과나무의 정보가 주어진다. 각 줄에는 각 줄기의 양 끝점의 번호와 줄기의 길이가 주어진다. 0번 지점은 새싹, 1~ N 번 지점은 사과를 의미하며 줄기의 길이는 1 이상 1,000,000 이하이다. 모든 줄기의 길이는 서로 다르다. 모든 줄기를 연결했을 때 새싹과 1~ N 번 사과가 모두 연결된다.

출력

첫 번째 줄에, 만들 수 있는 모든 사과나무의 광도의 합을 출력한다. 두 번째 줄에는 만들 수 있는 모든 사과나무의 에너지의 합을 출력한다. 답이 매우 클 수 있으므로 1,000,000,009로 나눈 나머지를 출력한다.

입출력 예제

| 입력 (stdin) | 출력 (stdout) |
|------------|-------------|
| 6 | |
| 0 1 3 | |
| 1 2 2 | |
| 2 3 5 | 1004 |
| 1 4 6 | 18026 |
| 4 5 7 | |
| 4 6 4 | |