격자 트리

0번부터 $N-1$번까지 번호가 매겨진 N개의 정점으로 이루어진 루트 있는 트리가 주어진다. 0번 정점이 이 트리의 루트이며, 모든 정점은 자식의 개수가 0개 혹은 2개이다. 자식이 2개인 정점은 왼쪽, 오른쪽 자식이 정해져 있다. 트리의 각 간선 e는 양의 정수 길이 c_e를 가진다.

우리는 이 트리를 2차원 좌표 평면 위에 그릴 것이다. 각 정점 v는 서로 다른 정수 격자점 $m_v = (x_v, y_v)$에 그린다. 이때 루트 정점은 $m_0 = (0,0)$에 그려야 한다. 정수 격자점은 x좌표와 y좌표가 모두 정수인 점을 뜻한다.

v와 v의 부모 p를 잇는 간선 $e = (p, v)$는 좌표 평면 상에서 m_p와 m_v를 잇는 경로로 그려진다. 이 경로는 다음 조건을 모두 만족해야 한다:

- m_p에서 m_v를 향해 경로를 따라 움직였을 때의 방향은 항상 x좌표가 증가하는 방향 혹은 y좌표가 증가하는 방향이어야 한다. 구체적으로, x좌표 및 y좌표가 모두 증가하는 방향은 불가능하다. 또한 정수 격자점에서만 방향을 전환할 수 있다. 즉, 경로의 길이가 k라고 하면 정확히 $k-1$개의 시점에만 그 방향을 바꿀 수 있다.
- v가 p의 왼쪽 자식이라면 경로는 m_p에서 x좌표가 증가하는 방향으로 출발해야 한다.
- v가 p의 오른쪽 자식이라면 경로는 m_p에서 y좌표가 증가하는 방향으로 출발해야 한다.
- 경로의 길이는 대응하는 간선의 길이 c_e 이상이어야 한다.
- 경로들은 **교차해서는 안 된다**: 다시 말해, 어떤 경로의 내부 점(시작점과 끝점이 아닌 점)은 다른 경로에 포함되어서는 안 된다.

트리 그리기에서 정점 v의 **깊이**를 $L(v) = x_v + y_v$로 정의하자. 우리가 그리게 될 트리에서 자식이 0개인 정점은 모두 같은 깊이를 가져야 한다. 이 깊이를 **격자 깊이**라고 하자.

모든 올바른 트리 그리기에 대해서 **격자 깊이**가 최소가 되는 경우를 구하여라.

함수 목록 및 정의

다음 함수를 구현해야 한다.

```
long long compute_min_depth(int N, vector<int> P, vector <int> C, vector<int> D)
```

- N: 정점의 개수
- P, C, D: 크기가 $N-1$인 정수 배열. 모든 $1 \le i \le N-1$ 에 대해, i번 정점의 부모는 $P[i-1]$번 정점이다. i번 정점과 부모를 잇는 간선을 e라 할 때 $c_e = C[i-1]$이다. $D[i-1] = 0$이면 i번 정점은 부모 정점의 왼쪽 자식이고, $D[i-1] = 1$이면 i번 정점은 부모 정점의 오른쪽 자식이다.
- 조건을 만족하는 트리 그리기는 항상 존재함을 증명할 수 있다. 이 함수는 그 중 격자 깊이의 최솟값을 반환해야 한다.
- 이 함수는 단 한 번만 호출된다.

제출하는 소스 코드의 어느 부분에서도 입출력 함수를 실행해서는 안 된다.

제약 조건

- 주어지는 간선들은 0번 정점을 루트로 하는 트리를 형성한다.
- 각 정점의 자식 개수는 0개이거나 2개이다.
- $3 \le N \le 200\,000$
- 모든 i에 대해 $0 \le P[i] \le N-1$ $(0 \le i \le N-2)$
- 모든 i에 대해 $1 \le C[i] \le 10^9$ $(0 \le i \le N-2)$
- 모든 i에 대해 $0 \le D[i] \le 1$ $(0 \le i \le N-2)$

부분문제

트리에서의 두 정점 사이의 거리를 두 정점을 잇는 유일한 경로를 구성하는 간선들의 길이 합이라 정의하자.

부분문제	점수	추가 조건
1	10	$N \le 7$
2	8	자식이 2개인 모든 정점 v에 대해 v의 자식 중 하나는 자식이 0개이다.
3	21	$N \le 5000$, 자식이 0개인 모든 정점 v에 대해 0번 정점과 v의 거리는 K (≤ 2500)로 동일하다.
4	29	$N \le 5000$, 모든 정점 v에 대해 0번 정점과 v의 거리는 2500 이하이다.
5	32	추가적인 제약조건이 없다.

채점 방식

부분문제 3은 격자 깊이가 정확히 K인 트리 그리기가 존재하지 않으면 `compute_min_depth`가 -1을 반환해도 정답으로 인정한다. 보다 정확하게는:

- 격자 깊이가 K인 트리 그리기가 **존재하는** 테스트 케이스:
  - K를 반환한 경우 만점을 받는다.
  - 그 외의 경우, 0점을 받는다.

- 격자 깊이가 K인 트리 그리기가 **존재하지 않는** 테스트 케이스:
  - 최소 격자 깊이를 반환한 경우, 만점을 받는다.
  - -1을 반환한 경우, 만점을 받는다.
  - 그 외의 경우, 0점을 받는다.

부분문제 3에서는 자식이 0개인 모든 정점 v에 대해 0번 정점과 v의 거리가 동일하며, 그 거리가 K임을 유념하자.

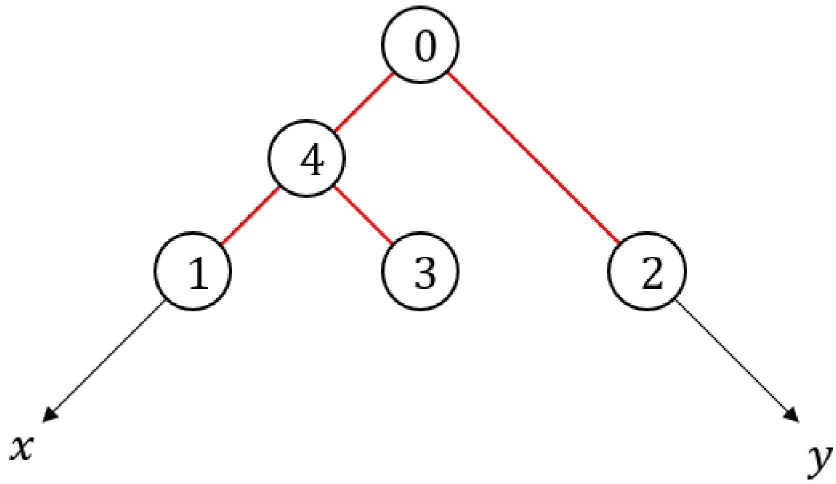
예제

예제 1

다음 호출을 생각해보자:

```
compute_min_depth(5, [4, 0, 4, 0], [1, 2, 1, 1], [0, 1, 1, 0])
```

- 아래 그림과 같이 격자 깊이가 2가 되도록 트리를 그릴 수 있다.



격자 깊이가 2 미만이 되도록 하는 트리 그리기가 존재하지 않음을 증명할 수 있다.

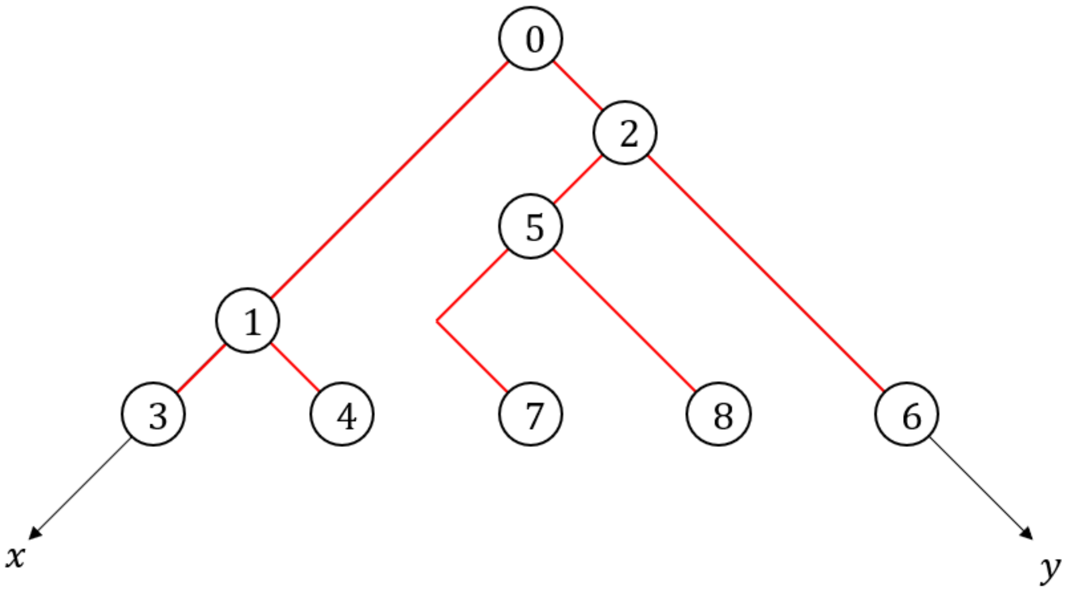
따라서 함수는 2를 반환해야 한다.

예제 2

다음 호출을 생각해보자:

```
compute_min_depth(9, [0, 0, 1, 1, 2, 2, 5, 5], [2, 1, 1, 1, 1, 1, 1, 1], [0, 1, 0,
1, 0, 1, 0, 1])
```

- 아래 그림과 같이 격자 깊이가 4인 트리를 그릴 수 있다.



따라서 함수는 4를 반환해야 한다.

샘플 그레이더

샘플 그레이더의 입력 형식은 다음과 같다.

- line 1: N
- 모든 $0 \le i < N-1$ 에 대해:
  - line $2+i$: $P[i]\ C[i]\ D[i]$

샘플 그레이더는 다음 형식으로 답을 출력한다:

- line 1: `compute_min_depth` 의 반환값