

## 문제 1. 샐러드바

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 128 megabytes

범수는 샐러드바에 갔다.  $n$ 개의 과일들이 일렬로 샐러드바에 있었다. 과일들은 사과나 오렌지이다. 범수는 샐러드를 만들 때 연속된 아무 구간이나 샐러드로 만들 수 있다.

그가 샐러드를 만들 때 고르는 과일은 왼쪽에서부터 오른쪽 혹은 오른쪽에서 왼쪽으로 샐러드에 추가된다. 범수는 오렌지를 좋아하기 때문에, 그가 만드는 샐러드에는 특별한 조건이 있다. 샐러드를 만드는 도중에, 왼쪽에서 오른쪽으로 만들든 오른쪽에서 왼쪽으로 만들든 오렌지의 수가 사과의 수 보다 적어서는 안된다. 범수가 조건을 만족하면서 샐러드로 만들 수 있는 가장 긴 구간을 구해주자.

### 입력 형식

첫째 줄에는 과일의 수를 의미하는 정수  $n$ 이 주어진다. ( $1 \leq n \leq 1,000,000$ ) 다음 줄에는  $n$ 개의 문자로 이루어진 문자열  $a_1, a_2, \dots, a_n$ 이 주어진다. ( $a_j \in \{j, p\}$ )  $a_i = j$ 이면  $i$ 번째 과일이 사과이고,  $a_i = p$ 이면  $i$ 번째 과일은 오렌지이다.

### 출력 형식

첫째 줄에 범수가 조건을 만족하면서 샐러드로 만들 수 있는 가장 긴 구간의 길이를 출력한다. 답이 0일 수 있음에 유의하여라.

### 서브태스크 1 (20 점)

- $n \leq 1,000$

### 서브태스크 2 (30 점)

- $n \leq 10,000$

### 서브태스크 3 (50 점)

추가 제한조건이 없다.

### 예제

standard input	standard output
6 jpjppj	4

### 참고 사항

가장 왼쪽과 오른쪽에 있는 사과를 제외하고 나머지 모든 과일들을 샐러드로 만들면 된다.

이 페이지는 빈 페이지입니다.

## 문제 2. 호텔

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 64 megabytes

지구이웨에는  $n$ 개의 마을이 있고,  $n-1$ 개의 도로로 연결되어 있다. 도로 하나는 두 마을을 직접 잇는다. 모든 도로는 같은 길이를 가지고 양방향 통행이 가능하다. 모든 마을에서 다른 마을까지 도로를 통해 이용하는 것이 가능하다. 즉 도로망은 트리 모양이다.

지구이웨의 왕인 범수는 세계의 여행자들을 끌어들이기 위한 세계의 럭셔리 호텔을 지으려고 한다. 범수는 호텔이 다른 마을에 있고, 서로가 서로로부터 같은 거리만큼 떨어져 있기를 원한다. 범수가 세 호텔을 세울 수 있는 경우의 수를 구하여라.

### 입력 형식

첫째 줄에는 마을의 수를 의미하는  $n$ 이 주어진다. ( $1 \leq n \leq 5,000$ ) 마을은 1부터  $n$ 까지 번호가 붙어있다. 지구이웨의 도로망은 다음  $n-1$ 개의 줄에 주어진다. 각 줄은 두개의 공백 하나로 구분된 정수  $a, b$ 가 있으며, 마을  $a$ 와 마을  $b$  사이에 직접 이어진 도로가 있다는 것을 의미한다. ( $1 \leq a \leq b \leq n$ )

### 출력 형식

첫째 줄에 가능한 호텔 배치의 가짓수를 정수 하나로 출력하여라.

### 서브태스크 1 (50 점)

- $n \leq 500$

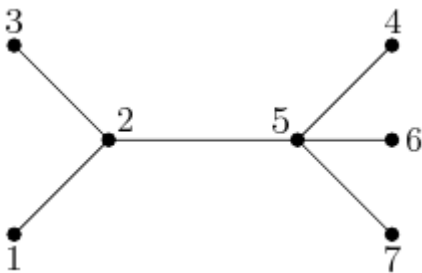
### 서브태스크 2 (50 점)

추가 제한조건이 없다.

### 예제

standard input	standard output
7 1 2 5 7 2 5 2 3 5 6 4 5	5

### 참고 사항



호텔을 세울 수 있는 위치는 다음과 같다:  $\{1, 3, 5\}$ ,  $\{2, 4, 6\}$ ,  $\{2, 4, 7\}$ ,  $\{2, 6, 7\}$ ,  $\{4, 6, 7\}$

이 페이지는 빈 페이지입니다.

## 문제 3. 블록

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 64 megabytes

어제 범수와 친구들은 유치원에서 다양한 색의 블록을 만들고 노느라 하루를 다 보냈다. 처음에 그들은 모형 빌딩을 만들었지만 곧 질려버렸다. 그래서 그들은 블록을 일렬로 배치하기로 했다. 단조로워 보이는 것을 막기 위해 같은 색의 두 블록을 옆에 놓지 않으려고 노력했다. 긴 시간 끝에 그들은 이 조건에 맞춰 블록을 놓는데에 성공했고, 부모님들은 다시 범수와 친구들을 집으로 데려갔다.

오늘 범수는 유치원에 일찍 왔다. 그는 어제 쌓은 블록이 계속 남아있다는 것에 만족했다. 하지만 그는 그 블록 위로 넘어지고 말았다. 범수는 그 블록들을 색별로 정리하고 다시 원래대로 복구하고 싶어 했다. 그가 기억한 것은 가장 왼쪽 끝과 오른쪽 끝의 블록 색 뿐이다.

범수에게 두 인접한 블록이 같은 색을 가지지 않고, 그가 기억하던 왼쪽 끝과 오른쪽 끝의 색을 유지하면서 어떻게 블록들을 배치해야 하는지 알려주자. 범수가 기억을 잘못하거나, 넘어지면서 블록들을 잃어버려서 새로 배치하는것이 불가능 할수도 있다.

### 입력 형식

첫째 줄에는 세 정수 블록 색의 가짓수  $k$ , 가장 왼쪽 끝 블록의 색  $p$ , 가장 오른쪽 끝 블록의 색  $q$ 가 공백 하나로 구분되어 주어진다. ( $1 \leq k \leq 1,000,000$ ,  $1 \leq p, q \leq k$ ) 둘째 줄에는  $k$ 개의 정수  $i_1, i_2, \dots, i_k$ 가 공백 하나로 구분되어 주어진다.  $i_j$ 는 색  $j$ 의 블록을 범수가 정확히  $i_j$ 개 가지고 있다는 것을 의미한다. 총 블록의 갯수는 백만개를 넘지 않는다. 즉,  $n = i_1 + i_2 + \dots + i_k \leq 1,000,000$  이다.

### 출력 형식

첫째 줄에  $n$ 개의 수를 공백 하나로 구분하여 출력하여라. 이 수는 가능한 블록 배치의 왼쪽 부터 오른쪽 까지의 색을 나타낸다. 만약 그런 배치가 없다면, 0을 출력하여라.

답이 여러개일 경우, 아무거나 출력하여도 좋다.

### 예제

standard input	standard output
3 3 1 2 3 3	3 2 1 3 2 3 2 1
3 3 1 2 4 2	0

### 참고 사항

첫번째 예제에서, 올바른 다른 정렬은 “3 1 2 3 2 3 2 1” 이다. 두번째 예제에서, 범수가 어딘가 실수를 해서 원래 조건을 만족하지 못했다.

이 페이지는 빈 페이지입니다.

## 문제 4. 택배

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 64 megabytes

범수는 컴퓨터게임을 파는 ZIG회사에서 일한다. 이 회사는 ZIG회사에서 판매되는 게임들을 공급하는 다른 게임회사와 같이 일한다. 범수는 ZIG회사와 택배회사와의 협력도 조사하고 있다. 그는 각 게임 패키지를 공급한 택배회사가 어느 회사인지에 대한 기록을 가지고 있다. 범수는 어떠한 회사도 불공평한 기회를 가지지 않았으면 한다.

만약 어떤 택배회사가 특정한 기간에 절반 초과의 패키지를 보냈다고 하면, 그 회사가 그 기간을 독점했다고 말한다. 범수는 어떤 기간에 특정 택배회사가 독점을 한 적이 있는지 알아보고 싶다.

범수를 도와주자! 독점하는 택배회사가 있는지 없는지 결정하는 프로그램을 작성하여라.

### 입력 형식

첫째 줄에는 ZIG회사가 배송한 게임 패키지의 수를 나타내는  $n$ 과, 게임 패키지를 배송한 날의 수를 나타내는  $m$ 이 공백 하나로 구분되어 들어온다. ( $1 \leq n, m \leq 500,000$ ) 택배 회사들은 1이상 최대  $n$ 이하의 수로 표현된다. 둘째 줄에는  $n$ 개의 정수  $p_1, p_2, \dots, p_n$ 이 공백 하나로 구분되어 들어온다. ( $1 \leq p_i \leq n$ )  $p_i$ 는  $i$  번째로 배송된 게임 패키지의 번호를 나타낸다. 다음  $m$ 개의 줄에는 쿼리가 한 줄에 하나씩 공백 하나로 구분된 두 수  $a, b$ 로 주어진다. ( $1 \leq a \leq b \leq n$ ) 이 쿼리는  $a$ 번째 패키지부터  $b$ 번째 패키지 까지 배송하는 기간에 독점하는 회사가 있는지 없는지 확인해야 한다. (이 기간은  $a$ 번째 패키지와  $b$ 번째 패키지를 포함한다.)

### 출력 형식

각 쿼리당 한 줄에 하나씩 답을 출력해야 한다. (총  $m$ 줄을 출력해야 한다.) 각 줄에는 정수 하나가 있어야 한다. 쿼리에 해당하는 구간에 독점하는 회사가 있다면 그 회사의 번호를, 없다면 0을 출력 해야 한다.

### 서브태스크 1 (30 점)

- $n, m \leq 5,000$

### 서브태스크 2 (35 점)

- $n, m \leq 50,000$

### 서브태스크 3 (35 점)

추가 제한조건이 없다.

### 예제

standard input	standard output
7 5	1
1 1 3 2 3 4 3	0
1 3	3
1 4	0
3 7	4
1 7	
6 6	

이 페이지는 빈 페이지입니다.



## 문제 5. 뱀

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 512 megabytes

뱀이  $3 \times n$  격자판을 채우고 있다. 뱀은 1부터  $3n$ 까지 번호가 붙은 구간들로 이루어져 있다. 하나의 숫자는 정사각형 하나를 차지하고, 연속된 숫자 (1과 2, 2와 3, 등등등...)가 차지하는 정사각형은 뱀을 공유한다. 다음은  $3 \times 9$  격자판을 채운 뱀의 예이다.:

7	6	5	4	17	18	19	20	21
8	1	2	3	16	15	26	25	22
9	10	11	12	13	14	27	24	23

뱀을 이루는 숫자들 몇개가 지워졌다. 원래 숫자를 복구하여라.

### 입력 형식

첫째 줄에는 격자판의 길이를 나타내는 정수  $n$ 이 주어진다. ( $1 \leq n \leq 1,000$ ) 다음 세개의 줄의  $i$ 번째 줄은 공백 하나로 구분된  $n$ 개의 정수가 주어진다.  $i$ 번째 줄의  $j$ 번째 숫자를  $a_{ij}$ 라고 할 때,  $a_{ij} > 0$ 인 경우 뱀의 번호를 나타내고  $a_{ij} = 0$ 인 경우 뱀의 번호를 알 수 없다는 것을 의미한다. ( $1 \leq a_{ij} \leq 3n$ )

### 출력 형식

세개의 줄을 출력해야 하고, 각 줄은  $n$ 개의 공백 하나로 구분된 정수여야 한다.  $3n$ 개의 숫자는 1부터  $3n$ 까지의 수의 순열이어야 한다. 출력은 올바른 뱀이어야 한다. 즉, 입력 데이터와 모순되지 않아야 하고, 뱀의 조건을 만족해야 한다.

올바른 뱀의 배치가 존재함을 가정해도 된다. 답이 여러개인 경우, 아무것이나 하나를 출력한다.

### 서브태스크 1 (15 점)

- $n \leq 10$

### 서브태스크 2 (25 점)

- $n \leq 40$

### 서브태스크 3 (30 점)

- $n \leq 300$

### 서브태스크 4 (30 점)

추가 제한조건이 없다.

## 예제

standard input	standard output
9	7 6 5 4 17 18 19 20 21
0 0 5 0 17 0 0 0 21	8 1 2 3 16 15 26 25 22
8 0 0 3 16 0 0 25 0	9 10 11 12 13 14 27 24 23
0 0 0 0 0 0 0 0 23	

## 문제 6. 카드

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 128 megabytes

$n$ 개의 카드가 특정한 순서로 테이블에 놓여 있다. 각 카드마다 앞면과 뒷면에 수가 하나씩 총 두개 쓰여있다. 처음에, 모든 카드는 앞면이 위로 가게 놓여있다. 위대한 마법사 범수는 그의 특기인 이분탐색 카드 마법을 여러번 쓰려고 한다. 하지만 마법을 쓰기 위해서는 수열에 보이는 숫자가 비내림차순이어야 한다. 즉, 범수는 몇몇 카드들을 뒤집어서 뒷면이 보이게 만들 것이다.

그리고, 그의 마법은 청중들이 참가해야 한다. 아앗, 범수의 마법이 실패하기를 원하는 경쟁자가 몇몇의 청중을 배치해 놓았다. 현장에 입장할 때 마다 이런 (수상한) 청중은 빛의 속도로 두 카드 두 장을 바꿀 것이다. 카드를 바꾼 이후에 범수는 자기가 원하는 카드들의 앞뒷면을 바꿀 수 있지만, 그의 위대한 이분탐색 카드 마법을 실행하지 못할 수도 있다. 그 때는 모자에서 토끼를 꺼내는 시시한 마법을 보여줄 수 밖에 없다. 각 카드가 바뀐 이후에 범수가 그의 위대한 마법을 실행할 수 있는지의 여부를 구하는 프로그램을 작성하여라.

### 입력 형식

첫째 줄에는 카드의 갯수를 의미하는 정수  $n$ 이 주어진다. ( $2 \leq n \leq 200,000$ ) 다음  $n$ 개의 줄에는 카드의 정보가 테이블에 놓여있는 순서대로 한 줄에 하나씩 주어진다.  $i$ 번째 줄에는 두 정수  $x_i$ 와  $y_i$ 가 공백 하나로 구분되어 들어온다. ( $0 \leq x_i, y_i \leq 10^7$ ). 이 수들은  $i$ 번째 카드에 적힌 숫자들을 나타낸다.:  $x_i$ 는 앞면,  $y_i$ 는 뒷면에 적힌 카드이다. 처음 카드 배치로 카드 마법을 사용하지 못할 수도 있다.

그 다음에 카드를 바꾸는 횟수  $m$ 이 주어진다. ( $1 \leq m \leq 1,000,000$ ) 다음  $m$ 개의 줄은 카드 교환을 의미한다.:  $j$ 번째 줄은  $a_j$ 와  $b_j$ 가 공백 하나로 구분되어 주어진다. ( $1 \leq a_j, b_j \leq n$ ) 이것은  $j$ 번째 (수상한) 청중이  $a_j$ 번째와  $b_j$ 번째 카드를 바꾼다는 것을 의미한다.

### 출력 형식

프로그램은  $m$ 개의 줄을 출력하여야 하고, 각 줄은 TAK이나 NIE여야 한다. 카드가 바뀐 이후에 범수가 카드 마술을 할 수 있으면 TAK, 아니면 NIE를 출력하여야 한다.

### 서브태스크 1 (30 점)

- $x_i = y_i$

### 서브태스크 2 (38 점)

- $n \leq 2,000$

### 서브태스크 3 (32 점)

추가 제한조건이 없다.

### 예제

standard input	standard output
4	2
2 5	3 4
3 4	1 3
6 3	
2 7	

이 페이지는 빈 페이지입니다.

## 문제 7. 범죄자

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 128 megabytes

지구이웨는 강을 낀 아름다운 마을이다. 강가에는  $n$ 개의 집이 있고, 상류에서 하류까지 1부터  $n$ 까지 번호가 붙은 집들이 있다. 지구이웨는 조용하고 모두가 행복한 마을이었다. 하지만 최근에 두 위험한 범죄자인 범수와 상수가 마을에 나타났다. 그들은 이미 도둑질을 많이 해서 사람들은 집을 나가는것도 무서워 했다.

범수와 상수가 하는 도둑질은 단순하지 않고, 전체 집을 도둑질 한다. 하나의 집을 나오면, 서로를 향해 걸어가고 뒤로 돌아가지는 않는다. 범수는 하류쪽으로(숫자가 큰 쪽으로), 상수는 상류쪽으로(숫자가 작은 쪽으로) 걸어간다. 길을 걷는 동안, 서로가 만나기 전에는 각자가 몇몇 집으로 들어가서 물건들을 훔친다. 그 후에 그들은 집에서 만나서 그들이 훔친 물건들을 나눠갖는다. 지구이웨 주민들은 현기증이 났다. - 그들의 평화를 되찾아야 한다! 그래서 탐정 사과에게 도움을 요청했다.

형사는 도둑이 같은 색의 집에 살고 있지만 어떤 사람인지는 모른다고 판정했다. 방금 익명의 사람이 도둑들이 급습중이라고 밝혔다. 소식통은 자신의 안전을 두려워해서, 어떤 집으로 들어갈지 밝히지는 않았다. 하지만 그들의 색은 밝혔다. 또한, 도둑들은 한 종류의 색의 집을 한 번만 도둑질 할 것이라는 미신을 가지고 있다.

사과는 더 이상 기다릴 수 없다. 그는 도둑들이 만나는 장소에서 잠복할 것이다. 도둑들이 모이는 것이 가능한 장소를 모두 찾는 프로그램을 작성하여라.

### 입력 형식

첫째 줄에는 집의 수를 의미하는 정수  $n$ 과 지구이웨에 있는 집의 색의 수를 의미하는 정수  $k$ 가 공백 하나로 구분되어 주어진다. ( $3 \leq n \leq 1,000,000$ ,  $1 \leq k \leq 1,000,000$ ,  $k \leq n$ ) 둘째 줄에는  $n$ 개의 정수로 이루어진 수열  $c_1, c_2, \dots, c_n$  ( $1 \leq c_i \leq k$ )가 공백 하나로 구분되어 주어진다. 이 수는 지구이웨에 있는 집들의 숫자들을 나열해 놓은 것이다.

셋째 줄에는 범수와 상수가 들어갈 집의 갯수를 나타내는  $m, l$ 이 공백 하나로 구분되어 주어진다. ( $1 \leq m, l \leq n$ ,  $m + l \leq n - 1$ ) 넷째 줄에는  $m$ 개의 정수  $x_1, x_2, \dots, x_m$ 이 공백 하나로 구분되어 주어지고, 범수가 (자신의 집을 제외하고) 들어갈 집의 색을 차례로 나타낸다. ( $1 \leq x_i \leq k$ ) 다섯째 줄에는  $l$ 개의 정수  $y_1, y_2, \dots, y_l$ 이 공백 하나로 구분되어 주어지고, 상수가 (자신의 집을 제외하고) 들어갈 집의 색을 차례로 나타낸다. ( $1 \leq y_i \leq k$ )  $x_m = y_l$ 이고, 이 수는 범수와 상수가 자신이 도둑질한 물건들을 나눌 장소를 나타낸다. (그들은 그 집도 도둑질 할 것이다.)

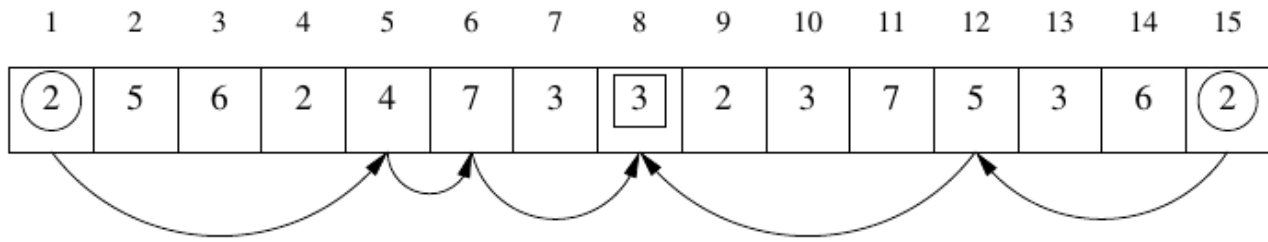
### 출력 형식

당신의 프로그램은 두개의 줄을 출력해야 한다. 첫째 줄은 위의 조건을 만족하면서 만날 수 있는 집의 갯수를 출력하여야 한다. 둘째 줄은 두 명이 만날 수 있는 집의 번호를 오름차순으로 출력해야 한다. 만약 상수와 범수가 만날 수 없다면, 첫째 줄은 0이어야하고, 둘째 줄은 비어있어야 한다.

### 예제

standard input	standard output
15 7	3
2 5 6 2 4 7 3 3 2 3 7 5 3 6 2	7 8 10
3 2	
4 7 3	
5 3	

### 참고 사항



위의 예제에서, 도둑들은 2번 색 (범수가 1번 혹은 4번 집, 상수가 15번 집) 혹은 6번 색 (범수가 3번 집, 상수가 14번 집)의 집에 살고 있을 것이다. 범수는 5번 집 (4번 색), 6번 집 (7번 색)을 차례로 들어가고 7번, 8번, 10번 집 (3번 색) 중의 하나에 들어갈 것이다. 상수는 12번 집 (5번 색)을 들어가고 7번, 8번, 10번 집 (3번 색) 중의 하나에 들어갈 것이다. 위의 그림은 범수가 1번집에 살고, 도둑들이 8번 집에서 모이는 상황을 나타낸다.

## 문제 8. 슈퍼컴퓨터

입력 파일: standard input  
출력 파일: standard output  
시간 제한: 3 seconds  
메모리 제한: 128 megabytes

범수는 새로운 구조의 슈퍼컴퓨터를 개발했다. 이 슈퍼 컴퓨터는 같은 종류의 처리장치 여럿으로 이루어져 있다. 하나의 처리장치는 1 단위시간에 하나의 명령을 실행할 수 있다.

이 컴퓨터용 프로그램은, 순차적인 실행이 아니라 트리모양의 구조를 가지고 있다. 하나의 명령은 0개, 1개 또는 2개 이상의 후속명령을 가질 수 있다.  $a$ 가  $b$ 의 후속명령이면,  $b$ 는  $a$ 의 부모명령이다.

프로그램의 명령들은 병렬처리가 될 수 있다. 그리고, 다양한 순서로 실행될 수 있다. 단 하나의 제약조건은, 어떤 명령은 그 부모명령이 실행되기 전에는 실행될 수 없다는 것이다. 예를 들어, 한 명령의 후속 명령들은 어떠한 순서로도 병렬로 실행될 수 있다.

범수는 실행해야 할 어떤 프로그램이 있다. 범수는 그의 자원을 효율적으로 사용하는 것을 좋아하기 때문에, 처리장치의 갯수가 프로그램의 실행에 어떠한 영향을 미칠지 궁금해 졌다. 그는 프로그램과 처리장치의 갯수가 주어졌을 때, 슈퍼컴퓨터로 프로그램을 실행하는 최소시간을 구하고 싶어 한다.

### 입력 형식

첫째 줄에는 범수의 프로그램의 명령어 수를 의미하는  $n$ 과 쿼리 갯수를 의미하는  $q$ 가 공백 하나로 구분되어 주어진다. ( $1 \leq n, q \leq 1,000,000$ )

둘째 줄에는  $q$ 개의 정수  $k_1, k_2, \dots, k_q$ 가 공백 하나로 구분되어 주어진다. ( $1 \leq k_i \leq 1,000,000$ )  $k_i$ 는 범수의  $i$ 번째 쿼리를 의미한다.

마지막 줄인 세번째 줄에는  $n - 1$ 개의 수열  $a_2, a_3, \dots, a_n$ 이 공백 하나로 구분되어 주어진다. ( $1 \leq a_i < i$ )  $a_i$ 번 명령은  $i$ 번 명령의 부모명령이다. 명령은 1부터  $n$ 까지 번호가 붙어있고, 1번 명령은 프로그램의 가장 처음 명령이다.

### 출력 형식

프로그램은  $q$ 개의 정수가 공백 하나로 구분된 한 줄을 출력해야 한다.  $i$ 번째 수는  $k_i$ 개의 처리장치로 이루어진 슈퍼컴퓨터가 명령을 실행하는데 걸리는 최소시간을 출력해야 한다.

### 서브태스크 1 (20 점)

- $n \leq 1,000$
- $q \leq 10$

### 서브태스크 2 (15 점)

- $n \leq 30,000$
- $q \leq 50$

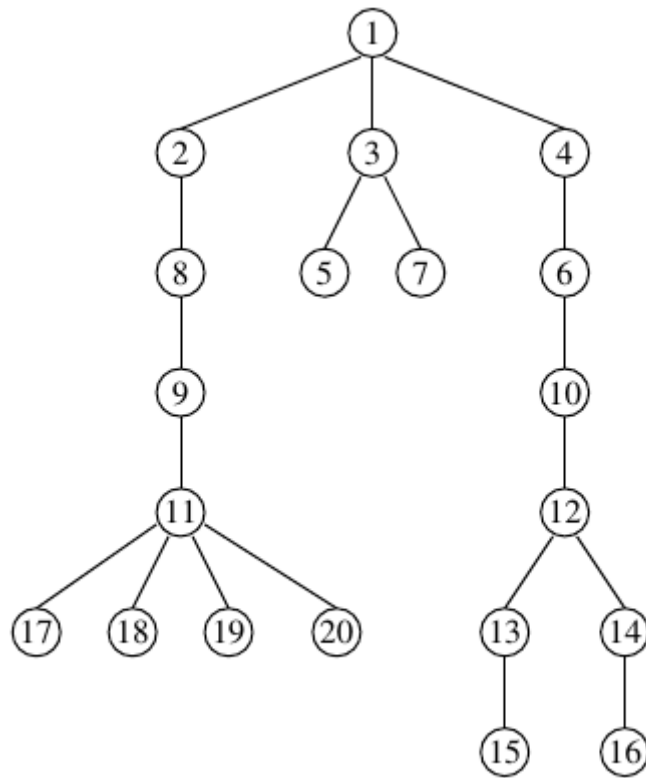
### 서브태스크 3 (65 점)

추가 제한조건이 없다.

### 예제

standard input	standard output
20 1 3 1 1 1 3 4 3 2 8 6 9 10 12 12 13 14 11 11 11 11	8

## 참고 사항



프로그램 명령은 다음과 같은 순서로 실행될 수 있다.

시간	명령			
1	1			
2	2	3	4	
3	5	6	7	
4	8	10		
5	9	12		
6	11	13	14	
7	15	16	17	
8	18	19	20	