

# Task: JOU

## Journey



Day 0. Source file `jou.*`

July 13, 2004

Available memory: 32 MB. Maximum running time: 0.5 s.

There are  $n$  cities in Byteland (numbered from 1 to  $n$ ), connected by bidirectional roads. The king of Byteland is not very generous, so there are only  $n - 1$  roads, but they connect the cities in such a way that it is possible to travel from each city to any other city.

One day, a traveller Byterider arrived in the city number  $k$ . He was planning to make a journey starting in the city  $k$  and visiting on his way cities  $m_1, m_2, \dots, m_j$  (not necessarily in this order) — the numbers  $m_i$  are all different and they are also different from  $k$ . Byterider — like every traveller — has only a limited amount of money, so he would like to visit all the cities that he has planned to visit using the shortest possible path (starting in the city  $k$ ). A path is one road or a sequence of roads, where every next road begins in the city where the previous one ends. Help Byterider to determine the length of the shortest path for his journey.

## Task

Write a program which:

- reads from the standard input:
  - the description of the roads connecting the cities of Byteland,
  - the number of the city where Byterider arrived,
  - a list of cities which Byterider would like to visit
- determines the minimum length of Byterider's journey,
- writes the result to the standard output

## Input

The first line of the standard input contains two integers  $n$  and  $k$  separated by a single space ( $2 \leq n \leq 50\,000$ ,  $1 \leq k \leq n$ ),  $n$  is the number of cities in Byteland and  $k$  is the number of the first city on Byterider's path. Each of the following  $n - 1$  lines contains the description of one road in Byteland. Line  $(i + 1)$  (for  $1 \leq i \leq n - 1$ ) contains three integers  $a_i, b_i$  and  $d_i$  separated by single spaces ( $1 \leq a_i, b_i \leq n$ ,  $1 \leq d_i \leq 1\,000$ ),  $a_i$  and  $b_i$  are the cities connected by the road, and  $d_i$  is the length of the road. Line  $(n + 1)$  contains one integer  $j$  — the number of cities which Byterider would like to visit ( $1 \leq j \leq n - 1$ ). The next line contains  $j$  different integers  $m_i$  separated by single spaces — the numbers of the cities that Byterider would like to visit ( $1 \leq m_i \leq n$ ,  $m_i \neq k$ ).

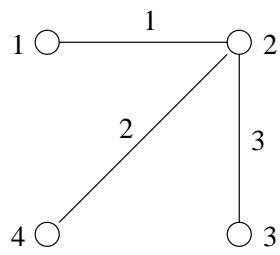
## Output

The first and only line of the standard output should contain exactly one integer: the length of the shortest path for Byterider's journey.

## Example

For the input data:

```
4 2
1 2 1
4 2 2
2 3 3
2
1 3
```



the correct result is:  
5