



## ふたつの通貨 (Two Currencies)

JOI 国には  $N$  個の都市があり, 1 から  $N$  までの番号が付けられている. また, JOI 国には  $N - 1$  本の道路があり, 1 から  $N - 1$  までの番号が付けられている. 道路  $i$  ( $1 \leq i \leq N - 1$ ) は都市  $A_i$  と都市  $B_i$  を双方向に結んでいる. どの都市からどの都市へも何本かの道路を通ることによって移動することができる.

JOI 国の道路には関所が設置されている. 関所は  $M$  個あり, 1 から  $M$  までの番号が付けられている. 関所  $j$  ( $1 \leq j \leq M$ ) は道路  $P_j$  上に設置されており, 通過するには通行料金として金貨 1 枚か銀貨  $C_j$  枚のどちらかを支払う必要がある.

JOI 国には  $Q$  人の国民がおり, 1 から  $Q$  までの番号が付けられている. 国民  $k$  ( $1 \leq k \leq Q$ ) は金貨  $X_k$  枚と銀貨  $Y_k$  枚を持っており, 都市  $S_k$  から都市  $T_k$  へ移動したい. また, 金貨は貴重なので, 国民は皆できるだけ多くの金貨を手元に残したい.

JOI 国の都市, 道路, 関所, 国民の情報が与えられたとき, それぞれの  $k$  ( $1 \leq k \leq Q$ ) について, 国民  $k$  が都市  $S_k$  から都市  $T_k$  へ移動できるか判定し, できる場合は最大で何枚の金貨を手元に残せるかを求めるプログラムを作成せよ.

### 入力

入力は以下の形式で標準入力から与えられる.

```
 $N$   $M$   $Q$   
 $A_1$   $B_1$   
 $A_2$   $B_2$   
:  
 $A_{N-1}$   $B_{N-1}$   
 $P_1$   $C_1$   
 $P_2$   $C_2$   
:  
 $P_M$   $C_M$   
 $S_1$   $T_1$   $X_1$   $Y_1$   
 $S_2$   $T_2$   $X_2$   $Y_2$   
:  
 $S_Q$   $T_Q$   $X_Q$   $Y_Q$ 
```



## 出力

標準出力に  $Q$  行出力せよ。  $k$  行目 ( $1 \leq k \leq Q$ ) には、国民  $k$  が都市  $S_k$  から都市  $T_k$  へ移動できる場合は手元に残せる金貨の枚数の最大値を、移動できない場合は  $-1$  を出力せよ。

## 制約

- $2 \leq N \leq 100\,000$ .
- $1 \leq M \leq 100\,000$ .
- $1 \leq Q \leq 100\,000$ .
- $1 \leq A_i \leq N$  ( $1 \leq i \leq N-1$ ).
- $1 \leq B_i \leq N$  ( $1 \leq i \leq N-1$ ).
- どの都市からどの都市へも何本かの道路を通ることによって移動することができる。
- $1 \leq P_j \leq N-1$  ( $1 \leq j \leq M$ ).
- $1 \leq C_j \leq 10^9$  ( $1 \leq j \leq M$ ).
- $1 \leq S_k \leq N$  ( $1 \leq k \leq Q$ ).
- $1 \leq T_k \leq N$  ( $1 \leq k \leq Q$ ).
- $S_k \neq T_k$  ( $1 \leq k \leq Q$ ).
- $0 \leq X_k \leq 10^9$  ( $1 \leq k \leq Q$ ).
- $0 \leq Y_k \leq 10^{18}$  ( $1 \leq k \leq Q$ ).
- 入力される値はすべて整数である。

## 小課題

1. (10 点)  $N \leq 2\,000$ ,  $M \leq 2\,000$ ,  $Q \leq 2\,000$ .
2. (28 点)  $C_1 = C_2 = \dots = C_M$ .
3. (30 点)  $A_i = i$ ,  $B_i = i + 1$  ( $1 \leq i \leq N - 1$ ).
4. (32 点) 追加の制約はない。



## 入出力例

入力例 1	出力例 1
5 4 3	1
1 2	2
1 3	-1
2 4	
2 5	
2 9	
2 4	
3 5	
4 7	
3 4 2 11	
5 3 4 5	
2 3 1 1	

国民 1 は以下のようにして、1 枚の金貨を残して都市 3 から都市 4 へ移動することができる。

1. 道路 2 を通り都市 3 から都市 1 へ移動する。道路 2 には関所 1, 2 があり、関所 1 で金貨 1 枚、関所 2 で銀貨 4 枚をそれぞれ支払って通過する。国民 1 が持っている金貨は 1 枚、銀貨は 7 枚になる。
2. 道路 1 を通り都市 1 から都市 2 へ移動する。道路 1 には関所はないので、国民 1 が持っている金貨は 1 枚、銀貨は 7 枚のままである。
3. 道路 3 を通り都市 2 から都市 4 へ移動する。道路 3 には関所 3 があり、関所 3 で銀貨 5 枚を支払って通過する。国民 1 が持っている金貨は 1 枚、銀貨は 2 枚になる。

国民 1 が最終的に 2 枚以上の金貨を持っているようにする方法は存在しないので、1 行目には 1 を出力する。

国民 2 は以下のようにして、2 枚の金貨を残して都市 5 から都市 3 へ移動することができる。

1. 道路 4 を通り都市 5 から都市 2 へ移動する。道路 4 には関所 4 があり、関所 4 で金貨 1 枚を支払って通過する。国民 2 が持っている金貨は 3 枚、銀貨は 5 枚になる。
2. 道路 1 を通り都市 2 から都市 1 へ移動する。道路 1 には関所はないので、国民 2 が持っている金貨は 3 枚、銀貨は 5 枚のままである。
3. 道路 2 を通り都市 1 から都市 3 へ移動する。道路 2 には関所 1, 2 があり、関所 1 で金貨 1 枚、関所 2 で銀貨 4 枚をそれぞれ支払って通過する。国民 2 が持っている金貨は 2 枚、銀貨は 1 枚になる。



国民 2 が最終的に 3 枚以上の金貨を持っているようにする方法は存在しないので、2 行目には 2 を出力する。

国民 3 は都市 2 から都市 3 へ移動できないので、3 行目には -1 を出力する。

この入力例は小課題 1,4 の制約を満たす。

入力例 2	出力例 2
10 7 9	3
1 8	6
6 3	6
5 9	7
7 9	7
3 1	3
3 4	1
10 1	2
2 6	2
5 6	
9 4	
7 4	
7 4	
2 4	
7 4	
7 4	
1 4	
8 6 5 3	
3 9 8 0	
4 7 6 15	
7 4 9 3	
6 4 8 0	
9 10 5 16	
5 3 2 4	
2 8 4 3	
6 1 3 3	

この入力例は小課題 1,2,4 の制約を満たす。



入力例 3	出力例 3
8 7 11	7
1 2	5
2 3	5
3 4	5
4 5	4
5 6	2
6 7	0
7 8	2
4 4	1
3 7	4
2 10	5
5 2	
4 1	
4 4	
5 6	
6 3 7 69	
7 1 5 55	
3 1 6 8	
8 2 5 45	
4 6 4 45	
6 1 3 33	
2 1 0 19	
3 7 2 31	
7 1 2 31	
7 2 4 58	
8 3 5 63	

この入力例は小課題 1,3,4 の制約を満たす。



入力例 4	出力例 4
8 7 11	1
1 8	3
1 4	1
3 1	7
3 6	0
6 7	4
2 1	5
5 2	7
5 5	8
5 8	10
4 7	6
6 6	
4 1	
6 4	
1 7	
4 7 2 18	
2 4 5 1	
4 2 1 32	
1 5 7 21	
2 5 0 50	
8 4 4 33	
1 7 6 16	
4 8 7 18	
1 2 8 13	
5 4 10 42	
7 1 6 40	

この入力例は小課題 1,4 の制約を満たす。