

## Tourists

Nom du problème	Tourists
Fichier d'entrée	entrée standard
Fichier de sortie	sortie standard
Limite de temps	4 secondes
Limite de mémoire	256 megaoctets

Il y a  $n$  villes dans l'Utopia, numérotées de 1 à  $n$ . Il y a aussi  $n - 1$  routes à double sens connectant ces villes. Il est possible de voyager entre chaque paire de villes en utilisant uniquement ces routes. Comme l'Utopia est magnifique, il y a  $m$  touristes, numérotés de 1 à  $m$ , qui sont actuellement en train de visiter ce pays. Initialement, le  $i^e$  touriste est en train de visiter la ville  $a_i$ . Il est possible que plusieurs touristes soient dans la même ville ; c'est-à-dire qu'il est possible que  $a_i = a_j$  pour une paire  $i, j$  telle que  $i \neq j$ .

Chaque touriste a une opinion sur le fait que sa visite d'Utopia soit intéressante ou non, représentée par un nombre. Initialement, l'opinion de chaque touriste est 0. Dans le but d'encourager plus de visites, le gouvernement utopien veut améliorer l'opinion des touristes sur le pays en organisant des événements dans certaines villes sélectionnées. Quand un événement a lieu dans une ville  $c$ , tous les touristes qui sont actuellement dans cette ville vont avoir leur opinion augmentée de  $d$ , où  $d$  est une valeur qui dépend du type d'événement.

Certains touristes ont prévu de voyager entre les villes durant leur séjour en Utopia. Même si voyager d'une ville à l'autre ne prend quasiment pas de temps (grâce aux routes utopiennes très efficaces), c'est toujours un désagrément et donc cela baisse l'opinion des touristes. Pour être exact, un touriste qui voyage en utilisant un chemin comprenant  $k$  routes différentes va voir son opinion baisser de  $k$  (les touristes choisissent toujours le chemin le plus court entre deux villes).

Le gouvernement utopien vous demande de traquer les opinions des touristes durant leurs voyages à travers le pays. Dans le cadre de cette demande, vous allez recevoir  $q$  requêtes faisant partie de l'entrée. Vous êtes supposée traiter et répondre aux requêtes dans l'ordre dans lequel elles apparaissent dans l'entrée.

## Entrée

La première ligne contient trois entiers  $n, m, q$  ( $2 \leq n \leq 200\,000$ ,  $1 \leq m, q \leq 200\,000$ ) - le nombre de villes, de touristes et de requêtes, respectivement.

La deuxième ligne contient  $m$  entiers  $a_1, a_2, \dots, a_m$  ( $1 \leq a_i \leq n$ ), où  $a_i$  représente la ville de départ du  $i^{\text{e}}$  touriste.

Les  $n - 1$  lignes suivantes contiennent 2 entiers chacune :  $v_i$  et  $w_i$  ( $1 \leq v_i, w_i \leq n$ ,  $v_i \neq w_i$ ) qui indiquent qu'il existe une route entre les villes  $v_i$  et  $w_i$ .

Les  $q$  lignes suivantes décrivent les requêtes dans l'ordre dans lequel elles sont demandées. Chaque ligne est de l'une des trois formes suivantes :

- La lettre 't' suivie de trois entiers  $f_i, g_i, c_i$  ( $1 \leq f_i \leq g_i \leq m$ ,  $1 \leq c_i \leq n$ ), signifiant que tous les touristes numérotés de  $f_i$  à  $g_i$  (les deux inclus) voyagent vers la ville  $c_i$ . Ceux qui sont déjà dans la ville  $c_i$  ne bougent pas, et leur opinion ne change pas.
- La lettre 'e' suivie de deux entiers  $c_i, d_i$  ( $1 \leq c_i \leq n$ ,  $0 \leq d_i \leq 10^9$ ), signifiant que dans la ville  $c_i$ , un événement a actuellement lieu, et il ajoute  $d_i$  à l'opinion des touristes de cette ville.
- La lettre 'q' suivie d'un entier  $v_i$  ( $1 \leq v_i \leq m$ ), représentant une question concernant l'opinion actuelle du touriste  $v_i$ .

Il est garanti qu'il existe au moins une requête 'q' dans l'entrée.

## Sortie

Afficher la réponse à chaque requête 'q', chacune sur une ligne séparée, dans l'ordre dans lequel elles ont été demandées.

## Score

Sous-tâche 1 (10 points) :  $n, m, q \leq 200$

Sous-tâche 2 (15 points) :  $n, m, q \leq 2\,000$

Sous-tâche 3 (25 points) :  $m, q \leq 2\,000$

Sous-tâche 4 (25 points) : Pas de requêtes 'e'

Sous-tâche 5 (25 points) : Aucune contrainte supplémentaire.

## Exemple d'entrée

8 4 11

1 4 8 1

6 4

6 3

3 7

6 5

5 1  
1 2  
1 8  
q 4  
t 3 4 5  
t 2 2 7  
q 4  
e 5 10  
e 1 5  
q 4  
t 1 1 5  
t 2 2 1  
q 1  
q 2

## Exemple de sortie

0  
-1  
9  
4  
-7