

KARWa 2024

Konkours d'algorithmique régionale Wallone



Problèmes

- A Antennes en pagailles
- B Bombe
- C Tonnelle
- D 24h vélo
- E Espion
- F Flot
- G Pairmutation
- H Hubs en folie
- I Ping Pong
- J Roller coaster
- K Kilomètres À Rallonge, Wagons en Avarie
- L Litige à tomates
- M Meurtrier



Copyright © 2024 by The KARWa 2024 jury. This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License.
<https://creativecommons.org/licenses/by-sa/4.0/>

A Antennes en pagailles

Limite de temps: 1s

La Komission des Antennes Régionales Wallones (KARWa) a besoin de vous! En effet, le gouvernement a investi dans énormément de nouveaux bâtiments partout en Wallonie. Nous sommes en 2024 donc, bien évidemment, ces bâtiments ont besoin d'une connexion internet fiable!

Pour ce faire, la Komission a acheté des antennes à mettre sur les bâtiments. Seulement, il y en a des meilleurs de disponibles... Ainsi, la Komission fait appel à vous afin de savoir où mettre chaque antenne pour avoir la meilleure couverture possible.



CC BY-SA 4.0 by X on iconvector

Pour réaliser cela, vous recevez une liste avec le besoin de connectivité de chaque bâtiment. Vous recevez aussi la liste des antennes à placer avec la connectivité qu'elles procurent.

Pour vous simplifier la tâche, votre superviseur vous dit qu'il est suffisant de maximiser la somme des produits entre la connectivité offerte par chaque antenne avec le besoin du bâtiment qui lui est relié. Par exemple, un bâtiment ayant un besoin de 100 et possédant une antenne offrant une connectivité de 33 aura un *score* de 3300. Il vous demande donc quel *score* vous pouvez obtenir pour chaque bâtiment dans le but que la somme de tous les scores soit la plus grande possible. Notez qu'une antenne ne peut être utilisée qu'une seule fois.

Entrée

L'entrée consiste en :

- Une ligne avec un entier n ($1 < N \leq 1000$), le nombre de bâtiments, qui est égal au nombre d'antennes.
- n lignes, chacune contenant un entier b_i ($1 \leq b \leq 1000$), le besoin de connectivité nécessaire au bâtiment i .
- n lignes, chacune contenant un entier c_i ($1 \leq c_i \leq 1000$), la connectivité offerte par l'antenne i .

Sortie

n lignes, où la i -ème ligne contient un entier s , le *score* qu'aura le bâtiment i dans une configuration optimale.

Exemple d'entrée 1

5
10
4
100
42
69
42
10
2
1
7

Exemple de sortie 1

20
4
4200
294
690

Exemple d'entrée 2

3
1
2
3
3
2
1

Exemple de sortie 2

1
4
9

B Bombe

Limite de temps: 1.5s

Oh non, il y a une bombe dans votre cuisine ! Vous devez la désamorcer avant qu'elle n'explose. Pour ce faire, vous devez résoudre un puzzle. Vous disposez d'une grille avec n lignes et m colonnes. Chaque case contient une couleur. Chaque ligne contient deux flèches, une qui pointe vers la gauche, l'autre vers la droite. Quand on appuie sur la flèche de droite, toutes les couleurs de la ligne sont déplacées d'une case vers la droite, et quand on appuie sur celle de gauche, elles sont déplacées d'une case vers la gauche. Les lignes sont cycliques, donc si une couleur sort d'un côté de la ligne, elle rentre de l'autre côté.



CC BY-SA 4.0 by
goff.brian on Vecteezy

Votre but est de trouver deux lignes qui sont égales après avoir entré une séquence de flèches. Si vous trouvez deux lignes qui sont égales, alors vous avez désamorcé la bombe. Si vous n'arrivez pas à trouver deux lignes qui sont égales, alors la bombe explose !

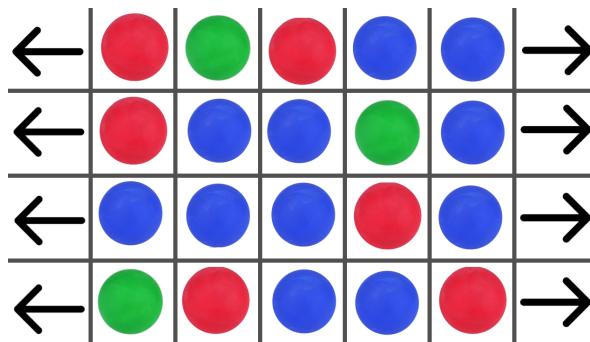


FIGURE B.1 – Illustration de l'exemple 1

Entrée

L'entrée consiste en :

- Une ligne avec deux entiers n et m ($0 \leq n, m \leq 10^4$), respectivement le nombre de lignes et le nombre de colonnes dans la grille,
- n lignes, chacune contenant une chaîne de caractères de longueur m . Le j -ème caractère de la i -ème ligne est $a_{i,j}$, la couleur de la cellule (i, j) .

Il est garanti que $a_{i,j}$ est une lettre de l'alphabet **majuscule ou minuscule**.

Il est aussi garanti que $n \times m \leq 5 \times 10^5$.

Sortie

L'indice des deux lignes qui sont égales en ordre croissant. S'il y a plusieurs paires, affichez celle qui a la plus petit premier indice. Si deux paires ont le même premier indice, affichez celle qui a le plus petit deuxième indice. Si vous ne trouvez pas deux lignes qui sont égales, affichez BOOM!.

Exemple d'entrée 1

4 5	1 4
RGRBB	
RBBGB	
BBBBRB	
GRBRR	

Exemple de sortie 1**Exemple d'entrée 2**

4 6	BOOM!
RRRRRR	
GGGGGG	
BBBBBB	
RGRGRG	

Exemple de sortie 2**Exemple d'entrée 3**

2 2	1 2
Aa	
aA	

Exemple de sortie 3**Exemple d'entrée 4**

2 16	1 2
RGRGRGRGRGRGRGRGR	
GRGRGRGRGRGRGRGR	

Exemple de sortie 4

C Tonnelle

Limite de temps: 1s

Le KARWa est fini, c'est l'heure des résultats! Suite à un bulletin météo favorable, les organisateurs ont décidé de faire l'annonce des résultats dehors. Sauf que le temps se gâte... Vite, il faut mettre tout le monde à l'abri! les organisateurs vous demandent de l'aide pour installer une tonnelle qui couvrira tout le monde. Ils vous donnent la position de toutes les personnes présentes, éparpillées sur la pelouse. Votre tâche consiste à calculer l'aire que doit faire la toile de la tonnelle afin de couvrir tout le monde. Dans un souci d'économie, vous devez trouver la plus petite toile rectangulaire possible. Nous considérerons qu'une personne est couverte par la tonnelle si elle est complètement en dessous, mais aussi si elle est placée pile sur un poteau de la tonnelle ou sur le bord de la toile. Pour simplifier, on considère une personne comme un point, sans largeur.



Une jolie tonnelle CC BY-SA 4.0 by X on Y

Entrée

L'entrée consiste en :

- Une ligne avec un seul entier n ($4 \leq n \leq 10^4$), le nombre de participants.
- n lignes, chacune contenant deux entiers x et y ($-500 \leq x, y \leq 500$), les coordonnées de chaque participant.

Sortie

L'aire de la plus petite tonnelle rectangulaire qui recouvre tout le monde. Votre réponse doit avoir une erreur absolue d'au plus 10^{-6} .

Exemple d'entrée 1

4
0 0
1 1
1 0
0 1

Exemple de sortie 1

1.0

Exemple d'entrée 2

6
3 0
1 6
-2 5
3 8
-1 -1
-2 -4

Exemple de sortie 2

60.0

Cette page est intentionnellement laissée vide.

D 24h vélo

Limite de temps: 1s

Après les 24h vélo, Alexandre fait ses comptes. Malheureusement, il ne se souvient plus de ce qu'il a acheté car il a perdu ses tickets de caisse. Heureusement, il se rappelle des prix des objets et du montant total qu'il a payé. Pouvez-vous retrouver quels objets il a achetés uniquement avec le montant total ? S'il n'existe aucune solution, retournez -1. Il est garanti que les objets sont uniques, et il n'est pas possible qu'Alexandre ait acheté plusieurs fois le même objet.

S'il existe plusieurs solutions, affichez-en une.



Entrée

L'entrée consiste en :

- une ligne avec deux entiers n et k ($0 \leq n \leq 64$, $0 \leq k \leq 3 \cdot 10^5$) représentant le nombre d'objets et le montant total payé,
- n lignes, chacune contenant s et p , où s est une chaîne de caractères de longueur au plus 10^4 donnant le nom d'un objet et p est son prix ($0 \leq p \leq 3 \cdot 10^5$).

Il est garanti que la somme des prix des objets est d'au plus $3 \cdot 10^5$.

Sortie

Sortez le nombre m d'objets qu'Alexandre a achetés, suivi de m lignes, chacune avec le nom d'un objet. Sortez -1 s'il n'y a aucun objet correspondant au montant total.

Exemple d'entrée 1

5 17
cara 5
eau 1
frites 3
pommes 10
pates 9

Exemple de sortie 1

3
cara
frites
pates

Exemple d'entrée 2

3 10000
sushi 1
poulet 1
coco 10000

Exemple de sortie 2

1
coco

Exemple d'entrée 3

3 9 lait 3 curry 3 persil 1	-1
--------------------------------------	----

Exemple de sortie 3**Exemple d'entrée 4**

8 1 a 1 b 1 c 1 d 1 e 1 f 1 g 1 h 1	1 b
---	--------

Exemple de sortie 4

E Espion

Limite de temps: 2s

Vous êtes un agent infiltré dans une organisation criminelle. En effet, un célèbre bandit nommé *John Windows* a capturé les membres du *Louvain-Li-Nux*. Pour ce faire, vous vous êtes infiltré dans l'endroit où ils sont capturés. Vous avez réussi à les libérer, mais vous devez vous échapper avant que les gardes ne vous attrapent. Vous avez t secondes pour vous échapper avant que votre hélicoptère s'envole.

Cependant, sur le chemin de la sortie se trouvent n gardes. Vous avez le choix entre les neutraliser ou les éviter. Vous possédez deux armes, un pistolet et un couteau. Le pistolet prend p_i secondes pour neutraliser le garde i , le couteau prend c_i secondes pour neutraliser le garde i et esquiver le garde i prend e_i secondes. Attention, vous ne pouvez pas neutraliser deux gardes à la suite avec la même arme ! Esquiver n'est pas considéré comme une arme. Dès lors, vous pouvez neutraliser un garde avec le pistolet, puis esquiver le prochain garde, puis neutraliser le suivant avec le pistolet.

Vous voulez savoir le nombre (x) maximum de gardes que vous pouvez neutraliser / éviter avant de vous fassiez attraper ainsi que le temps minimum qu'il vous faut pour échapper aux x gardes.



CC BY-SA 4.0 by Rizki Aditya Majid
on vecteezy

Entrée

L'entrée consiste en :

- une ligne avec deux entiers n et t ($1 \leq n \leq 5 \cdot 10^5$, $1 \leq t \leq 10^9$), le nombre de gardes et le temps que vous avez pour vous échapper,
- n lignes, la i -ème ligne contenant trois entiers p_i , c_i et e_i ($1 \leq p_i, c_i, e_i \leq 10^9$), le temps que prend chaque arme pour neutraliser le garde i .

Sortie

Sortez un entier x , le nombre de gardes que vous pouvez neutraliser avant de vous échapper et un entier m le temps minimum qu'il vous faut pour échapper aux x gardes.

Exemple d'entrée 1

4 14
3 4 5
2 2 2
8 1 10
100 1 10

Exemple de sortie 1

4 14

Exemple d'entrée 2

5 5
3 2 1
3 2 1
3 2 1
3 2 1
3 2 1

Exemple de sortie 2

5 5

Exemple d'entrée 3

7 6
1 2 3
2 1 3
1 2 3
3 2 1
1 2 3
2 1 3
10 10 10

Exemple de sortie 3

6 6

F Flot

Limite de temps: 1s

Vous êtes le coach de l'équipe de Belgique de course de bateau en papier sur l'eau ! La compétition consiste à déposer son bateau dans une rivière. Celui-ci se laissera porter par le courant de l'eau. Vous savez que les autres équipes sont très expérimentées, étant donné qu'ils ont des géologues qui analysent le terrain de fond en comble. Vous ne connaissez pas où la course va commencer et où elle va terminer. Cependant, vous connaissez les possibles endroits où la course peut commencer et où elle peut terminer.



CC BY-SA 4.0 par Luz Eugenia sur vecteezy

Dès lors, vous voulez savoir pour chaque arrivée potentielle, à quel départ vous devez déposer votre bateau pour que celui-ci arrive le plus vite possible !

De plus, cette compétition a une particularité des plus étranges : la longueur des courants est toujours égale à 5 mètres.

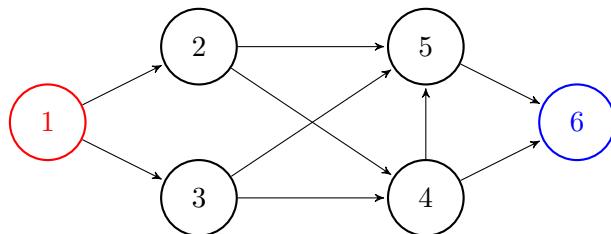


FIGURE F.1 – Illustration du test 1, la rivière rouge représente celle où l'on commence potentiellement et la rivière bleue représente celle où l'on termine potentiellement

Entrée

L'entrée consiste en :

1. une ligne avec deux entiers n et m ($2 \leq n \leq 10^5$, $0 \leq m \leq 10^5$), le nombre de rivières et le nombre de liens entre les rivières,
2. m lignes, chacune contenant 2 entiers u et v ($1 \leq u, v \leq n$, $1 \leq v \leq n$), indiquant qu'il y a un courant de la rivière u à la rivière v ,
3. une ligne avec un entier d ($0 \leq d \leq n$), le nombre de rivières où l'on peut commencer,
4. une ligne avec d entiers, le numéro des rivières où l'on peut commencer,
5. une ligne avec un entier a ($0 \leq a \leq n$), le nombre de rivières où l'on peut terminer,
6. une ligne avec a entiers, le numéro des rivières où l'on peut terminer la course.

Sortie

Pour chaque arrivée, une ligne contenant soit un entier, le numéro de la rivière de départ où l'on devrait commencer, soit -1 s'il n'y a aucune rivière qui mène à cette arrivée. Si plusieurs

rivières de départ ont la même distance, choisissez celle avec le plus petit indice.

Exemple d'entrée 1

6 9	1
1 2	
1 3	
2 4	
2 5	
3 5	
3 4	
4 5	
5 6	
4 6	
1	
1	
1	
6	

Exemple de sortie 1**Exemple d'entrée 2**

6 4	1
1 2	
2 3	
4 5	
5 6	
2	
1 4	
2	
3 6	

Exemple de sortie 2**Exemple d'entrée 3**

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

Exemple de sortie 3

Exemple d'entrée 4

2 0	
1	
1	
1	
2	

Exemple de sortie 4

-1

Cette page est intentionnellement laissée vide.

G Pairmutation

Dans le prestigieux **Komté Acculé** mais **Rétablit et Wizardesquement Abstrait** se tient chaque année un concours mémorable pour célébrer la paix avec le **Comté Presque Unanimement Masqué** mais **Observateur et Naturellement Supérieur**.

Cette année, vous avez l'honneur de pouvoir vous présenter et de montrer à tous que vous êtes le mage suprême. L'épreuve qui vous est assignée est la suivante :

Devant vous sont disposés n ballons numérotés de 1 à n . Ces ballons sont magiques et changent de place, mais vous réalisez rapidement qu'ils ne se déplacent pas de manière aléatoire ; ils suivent plutôt un schéma. En effet, ils se dirigent toujours vers la position associée à leur numéro.

Votre tâche consiste à déterminer le nombre de permutations que cette séquence présente. Heureusement, vous connaissez un sort qui fournit cette réponse, à condition que vous puissiez déterminer si cette séquence comporte un nombre pair ou impair de permutations.

Une permutation consiste en l'échange de positions entre deux ballons. Il est assuré que toute séquence peut être décomposée en un nombre fini de permutations de deux ballons.

Pour illustrer cela, prenons l'exemple concret suivant :

3 1 2

Dans cet exemple, le nombre de permutations est pair, car il suffit de permuter les ballons 1 et 2. Puis de permuter les ballons 2 et 3 pour que tous les ballons soient à leurs positions respectives.

Entrée

L'entrée consiste en :

- une ligne contenant un entier n ($1 \leq n \leq 10^6$) donnant la longueur de la séquence,
- une ligne contenant n entiers k_i ($1 \leq k_i \leq n$), chaque entier représentant le numéro du ballon à la position i . Il est garanti que chaque entier entre 1 et n apparaît une seule fois.

Sortie

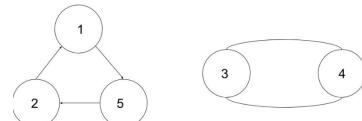
Le mot **Pair** si la séquence contient un nombre pair de permutations, **Impair** dans le cas contraire.

Exemple d'entrée 1

1 1	Pair
--------	------

Exemple de sortie 1

Limite de temps: 1s



Exemple de transposition

Exemple d'entrée 2

2 2 1	
----------	--

Exemple de sortie 2

	Impair
--	--------

Exemple d'entrée 3

3 3 1 2	
------------	--

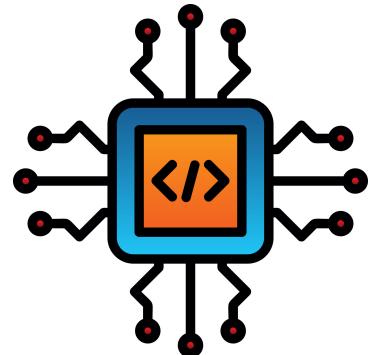
Exemple de sortie 3

	Pair
--	------

H Hubs en folie

Limite de temps: 1s

Le KARWa a tellement de succès cette année que, l'année prochaine, il sera nécessaire d'organiser n hubs pour accueillir les m participants. Les hubs sont situés aux coordonnées (a_i, b_i) , avec $i = 1, \dots, n$ et les participants aux coordonnées (x_j, y_j) , avec $j = 1, \dots, m$. Les organisateurs sont des fans du K-Means et afin de les aider à quantifier le placement des hubs, pouvez-vous déterminer la somme de toutes les distances euclidiennes au carré entre les hubs et les participants ?



CC BY-SA 4.0 by usmanabce on Vecteezy

Entrée

L'entrée consiste en :

- une ligne avec un entier n ($1 \leq n \leq 10^6$) représentant le nombre de hubs,
- n lignes, chacune contenant deux entiers a et b ($0 \leq a, b \leq 10^9$) représentant les coordonnées du hub,
- une ligne avec un entier m ($1 \leq m \leq 10^6$) représentant le nombre de participants,
- m lignes, chacune contenant deux entiers x et y ($0 \leq x_i, y_i \leq 10^9$) représentant la coordonnée du participant i .

Sortie

Un entier représentant la somme de toutes les distances euclidiennes au carré entre les hubs et les participants. Étant donné que la réponse peut être très grande, vous devez la donner modulo $10^9 + 7$.

Exemple d'entrée 1

```
2
2 2
5 5
2
2 5
6 6
```

Exemple de sortie 1

```
52
```

Exemple d'entrée 2

4
1 1
4 3
6 3
6 5
3
3 1
5 2
6 5

Exemple de sortie 2

131

Exemple d'entrée 3

8
2837 745
62 1162
2634 1112
1746 2618
847 127
986 1993
732 1273
2003 1998
4
1276 2231
1234 1234
287 2371
3000 3000

Exemple de sortie 3

110090622

Exemple d'entrée 4

5
283746263 475619273
987361523 361738847
281936352 666152443
143042069 482716253
1000000000 100000000
5
0 0
123456789 987654321
192837465 918273645
135792468 864209753
703692581 185296307

Exemple de sortie 4

391473143

I Ping Pong

Limite de temps: 1s

Vous êtes à la finale de la compétition de beer pong du Bénélux. Ce beer pong est un peu différent de celui que vous connaissez. Dans celui-ci, tous les verres sont alignés et espacés d'une distance x_i (en mètres). Vous voulez boire le plus de verres possible. Vous jouez aussi avec une balle un peu spéciale. C'est une balle qui fait des rebonds de distance constante en fonction de la force avec laquelle vous la lancez. C'est votre tour, vous avez une balle et vous devez choisir la force avec laquelle vous la lancez. Cependant, il est impossible de lancer la balle avec une force inférieure ou égale à 1 étant donné que ça demanderait une concentration hors norme. Vous voulez maximiser le nombre de verres que vous buvez. Pouvez-vous trouver la force optimale pour maximiser le nombre de verres que vous buvez ?



CC BY-SA 4.0 by Suprakon jutasuwan on Vecteezy



FIGURE I.1 – Exemple du sample 1

Entrée

L'entrée consiste en :

- une ligne avec un entier n ($1 \leq n \leq 10^6$), le nombre de gobelets - 1,
- n lignes, chacune comprenant un entier x_i ($1 \leq x_i \leq 10^6$), la distance entre les verres i et $i + 1$. À noter que le lanceur se trouve à la position 0 et que la première distance représente la distance entre le lanceur et le verre 1.

Il est garanti que la somme des x_i est inférieure ou égale à 10^6 .

Sortie

Sortez la force avec laquelle vous devez lancer la balle pour maximiser le nombre de verres que vous buvez. Ainsi que le nombre de verres que vous buvez.

Exemple d'entrée 1

```
5
1
1
1
3
3
```

Exemple de sortie 1

```
3 3
```

Exemple d'entrée 2

4	
5	
5	
5	
5	

Exemple de sortie 2

5	4
---	---

J Roller coaster

Limite de temps: 1s

Vous êtes invité par vos amis au célèbre parc d'attractions *KARWa land*. Bien que vous soyez un peu craintif vis-à-vis des attractions à grande vitesse depuis un accident d'enfance où vous avez chuté de 100 mètres, vous décidez de les accompagner pour ne pas les décevoir. Votre objectif est de profiter au maximum des attractions tout en évitant celles dont la vitesse est strictement supérieure à a .

Heureusement, le site Web du parc répertorie les attractions ainsi que la hauteur des rails. Chaque segment de rail est défini par deux points, et la vitesse dans ce segment est la différence de hauteur entre ces deux points. Bien sûr, ici on s'intéresse à la vitesse lorsque l'on descend étant donné que c'est à ce moment-là où l'on a le plus de sensations fortes.

Vous souhaitez déterminer combien d'attractions vous pouvez faire.



CC BY-SA 4.0 par commercialart sur vecteezy

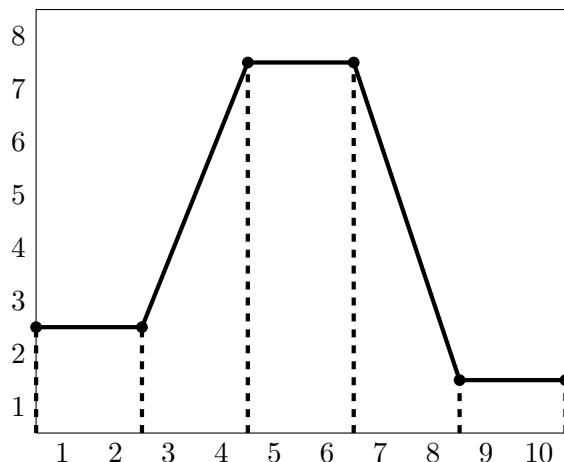


FIGURE J.1 – Illustration de l'exemple 1

Entrée

L'entrée consiste en :

- une ligne contenant deux entiers n et a ($0 \leq n \leq 1000$, $0 \leq a \leq 6 \cdot 10^6$), représentant respectivement le nombre d'attractions dans le parc et la vitesse maximale que vous pouvez supporter,
- n lignes, chacune contenant une chaîne de caractères s_i ($1 \leq |s_i| \leq 10$) et un entier m_i ($2 \leq m_i \leq 1000$), représentant le nom de l'attraction et le nombre de sommets dans l'attraction, suivis de m_i entiers $x_{i,j}$ ($0 \leq x_{i,j} \leq 10^6$) représentant la hauteur de chaque sommet.

Sortie

Soyez un entier k représentant le nombre d'attractions que vous pouvez faire, suivi de k chaînes de caractères représentant le nom de chaque attraction que vous pouvez faire.

Exemple d'entrée 1

1 6 poule 6 2 2 7 7 1 1	1 poule
----------------------------	------------

Exemple de sortie 1**Exemple d'entrée 2**

1 1 pomme 3 1 10 1	0
-----------------------	---

Exemple de sortie 2

K Kilomètres À Rallonge, Wagons en Avarie

Limite de temps: 2s

Vous souhaitez vous rendre à votre compétition d'algorithmique préférée qui commence ce soir ! Problème, vous vous y rendez en train, et la SNCB a tout juste déployé son plan *Kilomètres À Rallonge, Wagons en Avarie* (KARWA), visant à **maximiser le retard ferroviaire des belges**. Ce plan est très simple : certaines gares sont complètement imprévisibles et la SNCB peut vous emmener à un lieu non désiré, n'offrant aucun contrôle aux voyageurs de la destination. Heureusement pour vous, ce n'est pas encore le cas de toutes les gares.



La traversée de la Belgique s'avère donc compliquée. Chaque gare possède un certain nombre d'autres gares adjacentes (au moins une). Certaines gares appliquent déjà le plan KARWA, auquel cas votre train peut se retrouver à n'importe quelle gare adjacente, au bon vouloir de la SNCB, tandis que lorsque la gare n'applique pas le plan, c'est vous qui avez le choix de la destination à la gare suivante. Attention, certaines voies sont unidirectionnelles ! C'est-à-dire qu'il est possible que vous puissiez aller de A à B, mais pas de B à A.

Votre but est de minimiser le nombre de correspondances de votre point de départ à votre destination. Vous démarrez toujours de la gare 0 et devez atteindre la gare 1. Si vous avez une seule gare intermédiaire, par exemple $0 \rightarrow 2 \rightarrow 1$, alors vous aurez une seule correspondance.

Entrée

L'entrée consiste en :

- une ligne contenant deux entiers n ($2 \leq n \leq 10^4$) et m ($0 < m \leq n^2$), respectivement le nombre de gares belges sur le réseau et le nombre de liaisons entre deux gares,
- une ligne de n Booléens, la i -ème valeur vaut 1 si et seulement si la gare numéro i applique le plan KARWA,
- m lignes où chaque ligne contient deux entiers i, j ($0 \leq i, j < n$), indiquant que la gare numéro i a une voie lui permettant d'aller à la gare numéro j .

Sortie

Le plus petit nombre de correspondances afin de rejoindre la gare 1 en partant de la gare 0, ou "IMPOSSIBLE" si la SNCB peut ne jamais vous faire atteindre la gare 1.

Exemple d'entrée 1

4	6		
0	0	1	0
0	2		
2	1		
2	3		
3	0		
3	1		
1	0		

Exemple de sortie 1

2

Exemple d'entrée 2

3	4	
1	0	1
0	2	
0	1	
2	0	
1	0	

Exemple de sortie 2

IMPOSSIBLE

L Litige à tomates

Limite de temps: 1s

Après le carnage musical du KARWa-au-quai de l'année dernière, les organisateurs ont prévu un outil de sécurité au cas où un chanteur tourne à la catastrophe : des *tomates*. Pour cette année, chaque personne du public sera équipée de k tomates à lancer sur les chanteurs, ils sont libres de n'en lancer aucune, bien évidemment.

Un nouveau système de notation pour évaluer les performances des chanteurs a été mis en place. Chaque chanteur démarre à n points, et chaque tomate lancée par le public durant la performance diminue sa note de 1, jusqu'à un minimum de 0.

Vous chantez en dernier, et en fin stratège que vous êtes, vous avez été attentif au nombre de tomates lancées par le public et combien de spectateurs sont présents. Vous cherchez donc à connaître votre note minimale.



Entrée

L'entrée consiste en :

- une ligne contenant un entier n ($0 < n \leq 10^{16}$), la note maximale à laquelle démarrent les chanteurs,
- une ligne contenant deux entiers m et k ($0 < m, k \leq 10^9$), respectivement le nombre de spectateurs dans le public et le nombre de tomates que chaque spectateur reçoit,
- une ligne contenant un entier p ($0 \leq p \leq m \cdot k$), le nombre de tomates déjà lancées

Sortie

Un entier entre 0 et n , la note minimale que vous pouvez obtenir.

Exemple d'entrée 1

10	
12 2	
16	

Exemple de sortie 1

2	
---	--

Exemple d'entrée 2

5	
1 10	
2	

Exemple de sortie 2

0	
---	--

Cette page est intentionnellement laissée vide.

M Meurtrier

Limite de temps: 1s

Vous avez été appelé pour enquêter sur un crime qui a eu lieu dans un grand manoir. Quelqu'un a été tué! En tant que fin déTECTive que vous êtes, vous avez déjà émis une hypothèse : le meurtrier ne s'est pas fait voir par les autres membres de la maison avant le meurtre. Étant donné que pour planifier un meurtre, il faut être discret.

Vous savez également que le meurtrier a dû voir tout le monde pour planifier son meurtre (savoir si Albert est dans la cuisine ou si Michelle est dans sa chambre).

Vous pouvez interroger les n personnes qui étaient dans la maison au moment du meurtre.



CC BY-SA 4.0 par irkhamgram sur vecteezy

Interaction

Votre programme doit retourner le nom du meurtrier. Pour ce faire, vous allez interagir avec un programme.

En entrée, vous aurez une ligne contenant un entier $1 \leq n \leq 10^4$, le nombre de personnes. Ensuite, n lignes avec le nom des personnes présentes dans le manoir. Il est garanti que les noms des personnes sont distincts et contiennent uniquement des lettres alphanumériques. Il est aussi garanti que la taille des noms est au maximum de 1000.

Vous pouvez retourner une ligne avec l'un de ces mots :

- “? a b” demandez à a s'il a vu b .
- “! a” la personne a est le meurtrier.

L'interacteur vous répondra avec une ligne contenant l'un de ces mots :

- “OUI” si la personne a a vu la personne b .
- “NON” si la personne a n'a pas vu la personne b .

Une fois qu'une personne est accusée, votre programme devra se terminer. Vous pouvez poser au maximum $n - 1$ questions. Si vous en posez plus, votre programme sera terminé et sera considéré comme faux.

Lire	Interaction d'exemple	Écrire
		1
6 Alexis Bob Caroline David Eliot Francois		
	? Alexis Bob	
NON		
	? Bob Alexis	
OUI		
	? Caroline David	
OUI		
	? Eliot Francois	
OUI		
	! Eliot	

