

Maman j'ai raté la livraison!

Mutanguha est gérante d'un service de livraison automatique. Un robot se positionne aux coordonnées $(0,0)$ et se met à suivre une chaîne de N instructions $S[0], S[1], \dots, S[N-1]$. Chaque instruction correspond à un déplacement: à partir de la case (x, y) , l'instruction L le bouge vers $(x-1, y)$, R vers $(x+1, y)$, D vers $(x, y-1)$, et U vers $(x, y+1)$.

Lorsque ces instructions sont enregistrées, il n'est possible d'effectuer qu'un seul changement: annuler **un bloc contigu** d'instructions, qui peut être vide. Autrement dit, on peut choisir deux entiers l and r tels que $0 \leq l \leq r \leq N$ et remplacer S par $S[0], S[1], \dots, S[l-1], S[r], S[r+1], \dots, S[N-1]$. Les instructions sont ensuite effectués comme tels. Il y a également un coût associé à chaque instruction annulée: C_L pour L , C_R pour R , C_D pour D , C_U pour U .

Avant de démarrer le robot, Mutanguha découvre soudainement que le client l'attend à la case (X, Y) .

Aidez Mutanguha à trouver le coût minimal nécessaire pour corriger la trajectoire du robot, ou bien déterminez qu'il n'est pas possible de la corriger.

Détails d'Implémentation

Vous devez implémenter la procédure suivante:

```
int64 minimum_fee(string S, int32 X, int32 Y, int32 CL, int32 CR,
                  int32 CD, int32 CU)
```

- S : une chaîne de N caractères parmi LRDU , correspondante aux instructions initiales.
- X, Y : la position du client.
- CL, CR, CD, CU : le coût d'annulation pour les instructions $\text{L}, \text{R}, \text{D}, \text{U}$ respectivement.
- Cette procédure est appelée exactement une fois.
- Cette procédure doit retourner le coût minimum pour diriger le robot vers la case (X, Y) , ou bien -1 s'il n'est pas possible de le rediriger vers cette case.

Contraintes

- $1 \leq N \leq 200\,000$
- $-N \leq X, Y \leq N$

- $0 \leq C_L, C_R, C_D, C_U \leq 10^9$

Sous-tâches

Sous-tâche	Score	Contraintes additionnelles
1	15	$N \leq 2\,000$
2	10	$C_L = C_R = C_D = C_U = 0$
3	20	Il n'y a que des intructions R et U
4	20	Il n'y a que des instructions L et R; $Y = 0$
5	15	$C_L = C_R = C_D = C_U = 1$
6	20	Aucune contrainte additionnelle.

Exemples

Exemple 1

Considérons l'appel suivant:

```
minimum_fee("RUURLLL", -1, 0, 2, 3, 3, 4)
```

Cette chaîne d'instructions mènera le robot vers la case $(-1, 2)$, qui ne correspond pas à celle du client $(-1, 0)$.

Si Mutanguha annule le bloc UU, le robot exécute les instructions R, R, L, L, L et arrive à la position $(-1, 0)$. Le coût résultant est de $C_U + C_U = 8$.

Lorsque le bloc UURL est annulé, on aura le même résultat, or le coût total est supérieur au précédent: $4 + 4 + 3 + 2 = 13$. Puisqu'il n'est pas possible de corriger la trajectoire sans encourir de coût inférieur à 8, la procédure `minimum_fee` doit retourner 8.

Exemple 2

Considérons l'appel suivant:

```
minimum_fee("RR", 0, 1, 1, 1, 1, 1)
```

Puisque le robot n'effectue que des déplacements horizontaux, il n'y a aucun bloc qu'on pourra annuler pour qu'il se dirige vers $(0, 1)$. La procédure `minimum_fee` doit alors retourner -1 .

Exemple 3

Considérons l'appel suivant:

```
minimum_fee("LU", -1, 1, 7, 7, 7, 7)
```

Le robot arrive déjà vers $(-1,1)$, donc il est nécessaire d'annuler aucun bloc. La procédure `minimum_fee` doit alors retourner 0.

Évaluateur local

Format d'entrée:

```
N X Y
CL CR CD CU
S
```

Format de sortie:

```
F
```

Où F est la valeur retournée par `minimum_fee`.