

Hiéroglyphes

Une équipe de chercheurs étudie les similarités entre des séquences de hiéroglyphes. Ils représentent chaque hiéroglyphe par un entier positif ou nul. Pour effectuer leur étude, ils utilisent les concepts suivants sur les séquences.

Pour une séquence fixée A , une séquence S est une **sous-séquence** de A si et seulement si elle peut être obtenue en retirant des éléments (possiblement aucun) de A .

Le tableau ci-dessous montre quelques exemples de sous-séquences de la séquence $A = [3, 2, 1, 2]$.

Sous-séquence	Comment elle peut être obtenue à partir de A
$[3, 2, 1, 2]$	Aucun élément n'est retiré.
$[2, 1, 2]$	$[3, 2, 1, 2]$
$[3, 2, 2]$	$[3, 2, 4, 2]$
$[3, 2]$	$[3, 2, 4, 2]$ ou $[3, 2, 4, 2]$
$[3]$	$[3, 2, 4, 2]$
$[]$	$[3, 2, 4, 2]$

Au contraire, $[3, 3]$ ou $[1, 3]$ ne sont pas des sous-séquences de A .

Considérons deux séquences de hiéroglyphes A et B . Une séquence S est appelée une **sous-séquence commune** de A et B si et seulement si S est à la fois une sous-séquence de A et de B . De plus, on dit qu'une séquence U est une **sous-séquence commune universelle** de A et B si et seulement si les deux conditions suivantes sont vérifiées :

- U est une sous-séquence commune de A et B .
- Toutes les sous-séquences communes de A et B sont aussi des sous-séquences de U .

Il est possible de démontrer que deux séquences A et B ont au plus une seule sous-séquence commune universelle.

Les chercheurs ont trouvé deux séquences de hiéroglyphes A et B . La séquence A est composée de N hiéroglyphes et la séquence B est composée de M hiéroglyphes. Aidez les chercheurs à calculer la sous-séquence commune universelle de A et B , ou déterminez qu'il n'en existe pas.

Détails d'implémentation

Vous devez implémenter la fonction suivante.

```
std::vector<int> ucs(std::vector<int> A, std::vector<int> B)
```

- A : tableau de longueur N décrivant la première séquence.
- B : tableau de longueur M décrivant la deuxième séquence.
- S'il existe une sous-séquence commune universelle de A et B , la fonction doit renvoyer un tableau contenant cette séquence. Sinon, la fonction doit renvoyer $[-1]$ (un tableau de longueur 1, dont l'unique élément est -1).
- Cette fonction est appelée exactement une fois pour chaque test.

Contraintes

- $1 \leq N \leq 100\,000$
- $1 \leq M \leq 100\,000$
- $0 \leq A[i] \leq 200\,000$ pour tout i tel que $0 \leq i < N$
- $0 \leq B[j] \leq 200\,000$ pour tout j tel que $0 \leq j < M$

Sous-tâches

Sous-tâche	Score	Contraintes supplémentaires
1	3	$N = M$; A et B sont toutes les deux composées de N entiers distincts entre 0 et $N - 1$ (tous deux inclus)
2	15	Pour chaque entier k , (nombre d'éléments de A égal à k) + (nombre d'éléments de B égal à k) ≤ 3 .
3	10	$A[i] \leq 1$ pour tout i tel que $0 \leq i < N$; $B[j] \leq 1$ pour tout j tel que $0 \leq j < M$
4	16	Il existe une sous-séquence universelle de A et B .
5	14	$N \leq 3000$; $M \leq 3000$
6	42	Aucune contrainte supplémentaire.

Exemples

Exemple 1

Considérons l'appel de fonction suivant :

```
ucs([0, 0, 1, 0, 1, 2], [2, 0, 1, 0, 2])
```

Ici, les sous-séquences communes de A et B sont les suivantes : $[], [0], [1], [2], [0, 0], [0, 1], [0, 2], [1, 0], [1, 2], [0, 0, 2], [0, 1, 0], [0, 1, 2], [1, 0, 2]$ et $[0, 1, 0, 2]$.

Comme $[0, 1, 0, 2]$ est une sous-séquence commune de A et B , et que toutes les sous-séquences de A et B sont des sous-séquences de $[0, 1, 0, 2]$, la fonction doit renvoyer $[0, 1, 0, 2]$.

Exemple 2

Considérons l'appel de fonction suivant :

```
ucs([0, 0, 2], [1, 1])
```

Ici, la seule sous-séquence commune de A et B est la séquence vide $[]$. Par conséquent, la fonction doit renvoyer le tableau vide $[]$.

Exemple 3

Considérons l'appel de fonction suivant :

```
ucs([0, 1, 0], [1, 0, 1])
```

Ici, les sous-séquences communes de A et B sont : $[], [0], [1], [0, 1]$ et $[1, 0]$. Il peut être démontré qu'il n'existe pas de sous-séquence commune universelle. Donc, la fonction doit renvoyer $[-1]$.

Évaluateur d'exemple (sample grader)

Format d'entrée :

```
N M
A[0] A[1] ... A[N-1]
B[0] B[1] ... B[M-1]
```

Format de sortie :

```
T
R[0] R[1] ... R[T-1]
```

Ici, R est le tableau renvoyé par `ucs` et T est sa longueur.