

## 상형문자열

한 팀의 연구자들이 상형문자열 사이의 유사성을 연구하고 있다. 그들은 각각의 상형문자를 음이 아닌 정수로 표현한다. 그들이 연구에 사용하는 문자열 개념은 다음과 같다.

고정된 상형문자열  $A$ 에서 0개 이상의 상형문자를 지워서  $S$ 를 얻을 수 있으면  $S$ 를  $A$ 의 **부분수열 (subsequence)**이라고 부른다. 아래의 테이블은 상형문자열  $A = [3, 2, 1, 2]$ 의 부분수열의 예를 보여주고 있다.

| 부분수열         | $A$ 로부터 구하는 방법               |
|--------------|------------------------------|
| [3, 2, 1, 2] | 아무 원소도 삭제하지 않음               |
| [2, 1, 2]    | [3, 2, 1, 2]                 |
| [3, 2, 2]    | [3, 2, 1, 2]                 |
| [3, 2]       | [3, 2, 1, 2] or [3, 2, 1, 2] |
| [3]          | [3, 2, 1, 2]                 |
| [ ]          | [3, 2, 1, 2]                 |

반대로 [3, 3]와 [1, 3]는  $A$ 의 부분수열이 아니다.

두 상형문자열  $A$ 와  $B$ 를 생각해보자. 상형문자열  $S$ 가  $A$ 와  $B$  모두의 부분수열이면  $A$ 와  $B$ 의 **공통부분수열 (common subsequence)**이라고 부른다. 추가로, 상형문자열  $U$ 가 아래의 두 조건을 만족하면  $A$ 와  $B$ 의 **보편 공통부분수열 (universal common subsequence)**이라 부른다:

- $U$ 는  $A$ 와  $B$ 의 공통부분수열이다.
- 모든  $A$ 와  $B$ 의 공통부분수열은  $U$ 의 부분수열이다.

임의의 두 상형문자열  $A$ 와  $B$ 의 보편공통부분수열은 최대 1개라는 것은 보일 수 있다.

연구자들은 상형문자열  $A$ 와  $B$ 를 찾아냈다.  $A$ 의 길이는  $N$ 이고  $B$ 의 길이는  $M$ 이다. 여러분은 연구자들을 도와서  $A$ 와  $B$ 의 보편공통부분수열을 계산하던지 아니면 보편공통부분수열이 없다고 판정하라.

## Implementation details

여러분은 다음의 프로시저를 구현해야 한다.

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

- $A$ : 길이가  $N$ 인 첫번째 상형문자열
- $B$ : 길이가  $M$ 인 두번째 상형문자열
- $A$ 와  $B$ 의 보편공통부분수열이 존재하면, 이 프로시저는 보편공통부분수열을 저장하는 배열을 반환한다. 그렇지 않으면 이 프로시저는  $[-1]$ 을 반환한다. (이 배열은  $-1$ 만 원소로 가지는 길이가 1인 배열이다.).
- 이 프로시저는 각 테스트케이스에 대해서 오직 1번만 호출된다.

## 제약조건

- $1 \leq N \leq 100\,000$
- $1 \leq M \leq 100\,000$
- $0 \leq A[i] \leq 200\,000$  ( $0 \leq i < N$ )
- $0 \leq B[j] \leq 200\,000$  ( $0 \leq j < M$ )

## 서브태스크

| 서브태스크 | 점수 | 추가 제약조건   |
|-------|----|---|
| 1     | 3  | $N = M$ ; $A$ 와 $B$ 는 각각 0 이상 $N - 1$ 이하의 서로 다른 $N$ 개의 정수로 구성된다.  |
| 2     | 15 | 모든 정수 $k$ 는 $A$ 와 $B$ 에서 최대 3 번 등장한다. 다시 말해, 모든 정수 $k$ 에 대해 ( $A$ 에 포함된 $k$ 의 갯수) + ( $B$ 에 포함된 $k$ 의 갯수) 는 최대 3이다. |
| 3     | 10 | $A[i] \leq 1$ ( $0 \leq i < N$ ); $B[j] \leq 1$ ( $0 \leq j < M$ )  |
| 4     | 16 | $A$ 와 $B$ 의 보편공통부분수열이 존재한다.   |
| 5     | 14 | $N \leq 3000$ ; $M \leq 3000$   |
| 6     | 42 | 제약조건없음  |

## 예제

### 예제 1

다음의 호출을 생각해보자.

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

여기서  $A$ 와  $B$ 의 공통부분수열은 다음과 같다:  $[ ]$ ,  $[0]$ ,  $[1]$ ,  $[2]$ ,  $[0, 0]$ ,  $[0, 1]$ ,  $[0, 2]$ ,  $[1, 0]$ ,  $[1, 2]$ ,  $[0, 0, 2]$ ,  $[0, 1, 0]$ ,  $[0, 1, 2]$ ,  $[1, 0, 2]$ ,  $[0, 1, 0, 2]$ .

상형문자열  $[0, 1, 0, 2]$ 이  $A$ 와  $B$ 의 공통부분수열이고 모든  $A$ 와  $B$ 의 공통부분수열이  $[0, 1, 0, 2]$ 의 부분수열이므로 프로시저는  $[0, 1, 0, 2]$ 을 반환한다.

## 예제 2

다음의 호출을 생각해보자.

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

$A$ 와  $B$ 의 유일한 공통부분수열은 빈 상형문자열  $[]$ 이다. 따라서 프로시저는 빈 상형문자열 배열  $[]$ 을 반환한다.

## 예제 3

다음의 호출을 생각해보자.

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

$A$ 와  $B$ 의 공통부분수열은  $[], [0], [1], [0, 1]$  그리고  $[1, 0]$ 이다. 이 경우 보편공통부분수열이 존재하지 않고 프로시저는  $-1$ 을 반환한다.

## Sample Grader

입력 포맷:

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

출력 포맷:

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

$R$ 은  $\text{ucs}$ 가 반환하는 배열이고  $T$ 는 그 길이이다.