



## Zadanie F: Wyszukiwanie w ciągu przesuniętym cyklicznie

Twoim zadaniem jest wyszukiwanie podanych liczb w posortowanej tablicy, która została przesunięta cyklicznie. To znaczy, że oryginalny ciąg liczb został odcięty w pewnym momencie i powstałe dwie części zostały zamienione miejscami. W szczególności oryginalny ciąg jest również swoim przesunięciem cyklicznym. Przykładowo przesunięciem cyklicznym ciągu 1 2 3 4 5 jest 2 3 4 5 1, 5 1 2 3 4 lub 1 2 3 4 5.

Otrzymasz listę  $n$  liczb w kolejności opisanej jak powyżej oraz  $k$  pytań o pozycję liczby w tablicy, która została wcześniej podana (numerujemy od 1 do  $n$ ). Jeśli wyszukiwany jest element nieistniejący w tablicy, to powinniśmy zwrócić wartość  $-1$  pozycji. Każda liczba występuje w ciągu co najwyżej raz.

Uwaga: pomijając wczytanie ciągu, program powinien mieć złożoność  $k * \lg(n)$ .

### Format wejścia

W pierwszej linii standardowego wejścia znajdują się dwie liczby oddzielone spacją:  $n$  i  $k$  ( $1 \leq n, k \leq 1000000$ ). W kolejnej linii znajduje się  $n$  liczb całkowitych które najpierw były posortowane a następnie przesunięte cyklicznie - wejściowa tablica. W następnej znajduje się  $k$  oddzielonych spacjami liczb całkowitych - zapytania o indeksy. Liczby w tablicy i zapytania są na moduł nie większe niż 1000000.

### Format wyjścia

Na wyjściu powinno znaleźć się dokładnie  $k$  liczb oddzielonych spacjami - odpowiedzi na zapytania o indeksy. Jeżeli jakaś wartość nie istnieje w tablicy należy wypisać  $-1$ .

### Przykład

Dla danych wejściowych:

```
5 4
9 10 1 4 8
4 9 11 1
```

Poprawną odpowiedzią jest

```
4 5 -1 3
```