Problem 3237: Alt and Tab Simulation

Problem Information

Difficulty: Medium
Acceptance Rate: 51.42%
Paid Only: Yes
Tags: Array, Hash Table, Simulation

Problem Description

There are `n` windows open numbered from `1` to `n`, we want to simulate using alt + tab to navigate between the windows.

You are given an array `windows` which contains the initial order of the windows (the first element is at the top and the last one is at the bottom).

You are also given an array `queries` where for each query, the window `queries[i]` is brought to the top.

Return the final state of the array `windows`.

Example 1:

Input: windows = [1,2,3], queries = [3,3,2]

Output: [2,3,1]

Explanation:

Here is the window array after each query:

* Initial order: `[1,2,3]` * After the first query: `[_**3**_ ,1,2]` * After the second query: `[_**3**_ ,1,2]` * After the last query: `[_**2**_ ,3,1]`

Example 2:

Input: windows = [1,4,2,3], queries = [4,1,3]

Output: [3,1,4,2]

Explanation:

Here is the window array after each query:

* Initial order: `[1,4,2,3]` * After the first query: `[_**4**_ ,1,2,3]` * After the second query: `[_**1**_ ,4,2,3]` * After the last query: `[_**3**_ ,1,4,2]`

Constraints:

* `1 <= n == windows.length <= 105` * `windows` is a permutation of `[1, n]`. * `1 <= queries.length <= 105` * `1 <= queries[i] <= n`

Code Snippets

C++:

```
class Solution {
public:
vector<int> simulationResult(vector<int>& windows, vector<int>& queries) {

}
};
```

Java:

```
class Solution {
public int[] simulationResult(int[] windows, int[] queries) {

}
}
```

Python3:

```
class Solution:
def simulationResult(self, windows: List[int], queries: List[int]) ->
List[int]:
```