Problem 1409: Queries on a Permutation With Key

Problem Information

Difficulty: Medium
Acceptance Rate: 84.72%
Paid Only: No
Tags: Array, Binary Indexed Tree, Simulation

Problem Description

Given the array `queries` of positive integers between `1` and `m`, you have to process all `queries[i]` (from `i=0` to `i=queries.length-1`) according to the following rules:

* In the beginning, you have the permutation `P=[1,2,3,...,m]`. * For the current `i`, find the position of `queries[i]` in the permutation `P` (**indexing from 0**) and then move this at the beginning of the permutation `P`. Notice that the position of `queries[i]` in `P` is the result for `queries[i]`.

Return an array containing the result for the given `queries`.

Example 1:

Input: queries = [3,1,2,1], m = 5 **Output:** [2,1,2,1] **Explanation:** The queries are processed as follow: For i=0: queries[i]=3, P=[1,2,3,4,5], position of 3 in P is **2** , then we move 3 to the beginning of P resulting in P=[3,1,2,4,5]. For i=1: queries[i]=1, P=[3,1,2,4,5], position of 1 in P is **1** , then we move 1 to the beginning of P resulting in P=[1,3,2,4,5]. For i=2: queries[i]=2, P=[1,3,2,4,5], position of 2 in P is **2** , then we move 2 to the beginning of P resulting in P=[2,1,3,4,5]. For i=3: queries[i]=1, P=[2,1,3,4,5], position of 1 in P is **1** , then we move 1 to the beginning of P resulting in P=[1,2,3,4,5]. Therefore, the array containing the result is [2,1,2,1].

Example 2:

Input: queries = [4,1,2,2], m = 4 **Output:** [3,1,2,0]

Example 3:

Input: queries = [7,5,5,8,3], m = 8 **Output:** [6,5,0,7,5]

Constraints:

* `1 <= m <= 10^3` * `1 <= queries.length <= m` * `1 <= queries[i] <= m`

Code Snippets

C++:

```
class Solution {
public:
vector<int> processQueries(vector<int>& queries, int m) {

}
};
```

Java:

```
class Solution {
public int[] processQueries(int[] queries, int m) {

}
}
```

Python3:

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