The next greater element of some element x in an array is the first greater element that is to the right of x in the same array.

You are given two distinct 0-indexed integer arrays nums1 and nums2, where nums1 is a subset of nums2.

For each 0 <= i < nums1.length, find the index j such that nums1[i] == nums2[j] and determine the next greater element of nums2[j] in nums2. If there is no next greater element, then the answer for this query is -1.

Return *an array* ans *of length* nums1.length *such that* ans[i] *is the next greater element as described above.*

Example 1:

Input: nums1 = [4,1,2], nums2 = [1,3,4,2]
Output: [-1,3,-1]
Explanation: The next greater element for each value of nums1 is as follows:
- 4 is underlined in nums2 = [1,3,4,2]. There is no next greater element, so the answer is -1.
- 1 is underlined in nums2 = [1,3,4,2]. The next greater element is 3.
- 2 is underlined in nums2 = [1,3,4,2]. There is no next greater element, so the answer is -1.

Example 2:

Input: nums1 = [2,4], nums2 = [1,2,3,4]
Output: [3,-1]
Explanation: The next greater element for each value of nums1 is as follows:
- 2 is underlined in nums2 = [1,2,3,4]. The next greater element is 3.
- 4 is underlined in nums2 = [1,2,3,4]. There is no next greater element, so the answer is -1.

Constraints:

- **1 <= nums1.length <= nums2.length <= 1000**
- **0 <= nums1[i], nums2[i] <= 10^4**
- **All integers in nums1 and nums2 are unique.**
- **All the integers of nums1 also appear in nums2.**

Follow up: Could you find an O(nums1.length + nums2.length) solution?

Solution:

```
class Solution {
    public int[] nextGreaterElement(int[] nums1, int[] nums2) {
         Map<Integer, Integer> map = new HashMap();
        for(int i = 0; i < nums2.length; i++) {
            for(int j = i; j < nums2.length; j++) {
                if(nums2[j] > nums2[i]) {
                    map.put(nums2[i], nums2[j]);
                    break;
                } else if(j == nums2.length - 1) {
                    map.put(nums2[i], -1);
                }
            }
        }
        for(int i = 0; i < nums1.length; i++) {
            nums1[i] = map.get(nums1[i]);
        }
        return nums1;
    }
}
```