Intersection of Two Arrays

Problem condition:

Given two integer arrays nums1 and nums2, return *an array of their intersection*.
Each element in the result must be **unique** and you may return the result in **any order**.

The intersection of two arrays is defined as the set of elements that are present in both arrays.

Algorithm:

At first create an empty array that we will return at the end (resultArray = [])

Then we have to go through the entire array nums1 and check two conditions:
1) Element of nums1 must be in the nums2 array
2) Element of nums1 must not be in the result array

If both conditions are true, then we add current element of nums1 to result array and check next element

Example nums1 = [1, 2, 3, 3, 4], nums2 = [3, 5, 4]:

Iterations:

1) nums1[i] = 1
Element of nums1 must be in the nums2 array - false
Element of nums1 must not be in the result array - false
Do nothing, resultArray = []

2) nums1[i] = 2
Element of nums1 must be in the nums2 array - false
Element of nums1 must not be in the result array - false
Do nothing, resultArray = []

3) nums1[i] = 3
Element of nums1 must be in the nums2 array - true
Element of nums1 must not be in the result array - true
Add 3 to the result array, resultArray = [3]

4) nums1[i] = 3
Element of nums1 must be in the nums2 array - true
Element of nums1 must not be in the result array - false
Do nothing, resultArray = [3]

5) nums1[i] = 4
Element of nums1 must be in the nums2 array - true
Element of nums1 must not be in the result array - true
Add 4 to the result array, resultArray = [3, 4]

The array nums1 has ended, return resultArray