**[Minimum Common Value](https://leetcode.com/problems/minimum-common-value/)**

Given two integer arrays nums1 and nums2, sorted in non-decreasing order, return *the****minimum integer common****to both arrays*. If there is no common integer amongst nums1 and nums2, return -1.

Note that an integer is said to be **common** to nums1 and nums2 if both arrays have **at least one** occurrence of that integer.

**Example 1:**

**Input:** nums1 = [1,2,3], nums2 = [2,4]

**Output:** 2

**Explanation:** The smallest element common to both arrays is 2, so we return 2.

**Example 2:**

**Input:** nums1 = [1,2,3,6], nums2 = [2,3,4,5]

**Output:** 2

**Explanation:** There are two common elements in the array 2 and 3 out of which 2 is the smallest, so 2 is returned.

**Constraints:**

* 1 <= nums1.length, nums2.length <= 105
* 1 <= nums1[i], nums2[j] <= 109
* Both nums1 and nums2 are sorted in **non-decreasing** order.

class Solution {

public:

    int getCommon(vector<int>& nums1, vector<int>& nums2) {

        int common = INT\_MAX;

        int i = 0, j = 0;

        while (i < nums1.size() && j < nums2.size()) {

            if (nums1[i] == nums2[j]) {

                common = nums1[i];

                break;

            } else if (nums1[i] < nums2[j]) {

                i++;

            } else {

                j++;

            }

        }

        return common != INT\_MAX ? common : -1;

    }

};

<https://leetcode.com/problems/minimum-common-value/?envType=daily-question&envId=2024-03-09>