# 题目

给你两个整数数组nums1和nums2，它们已经按非降序排序，请你返回两个数组的最小公共整数。如果两个数组nums1和nums2没有公共整数，请你返回-1。

如果一个整数在两个数组中都 至少出现一次，那么这个整数是数组nums1和nums2公共的。

示例 1：

输入：nums1 = [1,2,3], nums2 = [2,4]

输出：2

解释：两个数组的最小公共元素是 2 ，所以我们返回 2 。

示例 2：

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

输出：2

解释：两个数组中的公共元素是 2 和 3 ，2 是较小值，所以返回 2 。

提示：

1 <= nums1.length, nums2.length <= 105

1 <= nums1[i], nums2[j] <= 109

nums1 和 nums2 都是非降序的。

# 分析

## 方法一：双指针

分析：

由于数组已经排好序了，我们用两个指针分别指向 nums1 和 nums2 中的第一个数。后续哪个数较小就让哪个指针前进，直到两个指针指向的数都一样。

代码：

class Solution {

public:

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

int n = nums1.size(), m = nums2.size();

for (int i = 0, j = 0; i < n && j < m; ) {

if (nums1[i] == nums2[j]) return nums1[i];

else if (nums1[i] < nums2[j]) i++;

else j++;

}

return -1;

}

};

复杂度分析：

时间复杂度：复杂度O(n+m)。

空间复杂度：O(1)。