class Solution:

    def findMedianSortedArrays(

        self, nums1: List[int], nums2: List[int]

    ) -> float:

        m, n = len(nums1), len(nums2)

        p1, p2 = 0, 0

        # Get the smaller value between nums1[p1] and nums2[p2].

        def get\_min():

            nonlocal p1, p2

            if p1 < m and p2 < n:

                if nums1[p1] < nums2[p2]:

                    ans = nums1[p1]

                    p1 += 1

                else:

                    ans = nums2[p2]

                    p2 += 1

            elif p2 == n:

                ans = nums1[p1]

                p1 += 1

            else:

                ans = nums2[p2]

                p2 += 1

            return ans

        if (m + n) % 2 == 0:

            for \_ in range((m + n) // 2 - 1):

                \_ = get\_min()

            return (get\_min() + get\_min()) / 2

        else:

            for \_ in range((m + n) // 2):

                \_ = get\_min()

            return get\_min()