

4. Median of Two Sorted Arrays

Hard

Given two sorted arrays nums1 and nums2 of size m and n respectively, return the median of the two sorted arrays.

The overall run time complexity should be $O(\log(m+n))$.

Example 1:

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

Output: 2.00000

Explanation: merged array = [1,2,3] and median is 2.

Example 2:

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

Output: 2.50000

Explanation: merged array = [1,2,3,4] and median is $(2 + 3) / 2 = 2.5$.

B



A



```
def findMedianSortedArrays(nums1: List[int], nums2: List[int]) -> float:
    A, B = nums1, nums2
    total = len(nums1) + len(nums2)
    half = total // 2

    if len(B) < len(A):
        A, B = B, A

    left, right = 0, len(A) - 1

    while True:
        i = (left + right) // 2
        j = half - i - 2

        Aleft = A[i] if i >= 0 else float("-infinity")
        Aright = A[i + 1] if (i + 1) < len(A) else float("infinity")
        Bleft = B[j] if j >= 0 else float("-infinity")
        Bright = B[j + 1] if (j + 1) < len(B) else float("infinity")

        if Aleft <= Bright and Bleft <= Aright:
            if total % 2:
                return min(Aright, Bright)
            return (max(Aleft, Bleft) + min(Aright, Bright)) / 2
        elif Aleft > Bright:
            right = i - 1
        else:
            left = i + 1
```