30. 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)).

Program:

```
def findMedianSortedArrays(nums1, nums2):
    if len(nums1) > len(nums2):
        nums1, nums2 = nums2, nums1
    m, n = len(nums1), len(nums2)
    total_len = m + n
    half_len = (total_len + 1) // 2
    left, right = 0, m
    while left <= right:
        i = (left + right) // 2
        j = half_len - i
        if i < m and nums1[i] < nums2[j - 1]:
            left = i + 1
        elif i > 0 and nums1[i - 1] > nums2[j]:
            right = i - 1
        else:
            if i == 0:
                left_max = nums2[j - 1]
            elif j == 0:
                left_max = nums1[i - 1]
            else:
                left_max = max(nums1[i - 1], nums2[j - 1])
            if total_len % 2 == 1:
                return left_max
            if i == m:
                right_min = nums2[j]
            elif j == n:
                right_min = nums1[i]
            else:
```

```
            right_min = min(nums1[i], nums2[j])
        return (left_max + right_min) / 2
nums1 = [1, 3]
nums2 = [2]
print(findMedianSortedArrays(nums1, nums2))  # Output: 2.0
nums1 = [1, 2]
nums2 = [3, 4]
print(findMedianSortedArrays(nums1, nums2))
```

output:

```
6 3 2 1 5 0
=== Code Execution Successful ===
```

time complexity:O(log(min(m,n)))