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):
    m, n = len(nums1), len(nums2)
    if m > n:
        nums1, nums2, m, n = nums2, nums1, n, m
    low, high = 0, m
    while low <= high:
        partitionX = (low + high) // 2
        partitionY = (m + n + 1) // 2 - partitionX
        maxLeftX = float('-inf') if partitionX == 0 else nums1[partitionX - 1]
        minRightX = float('inf') if partitionX == m else nums1[partitionX]
        maxLeftY = float('-inf') if partitionY == 0 else nums2[partitionY - 1]
        minRightY = float('inf') if partitionY == n else nums2[partitionY]
        if maxLeftX <= minRightY and maxLeftY <= minRightX:
            if (m + n) % 2 == 0:
                return (max(maxLeftX, maxLeftY) + min(minRightX, minRightY)) / 2
            else:
                return max(maxLeftX, maxLeftY)
        elif maxLeftX > minRightY:
            high = partitionX - 1
        else:
            low = partitionX + 1

nums1 = [1, 3]
nums2 = [2]
print(findMedianSortedArrays(nums1, nums2))
```

OUTPUT: 


TIME COMPLEXITY: O ((m+n)log(m+n))