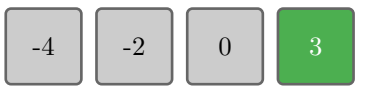
Kth Smallest Product of Two Sorted Arrays

Frame 110: Process

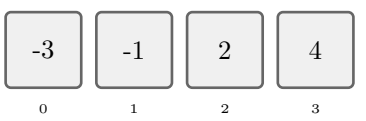
`low=-5, high=-3, mid=-4, count=4, current_num1=3, threshold=-1.33`



Pseudocode:

```
def kthSmallestProduct(nums1, nums2, k):
    #
    low = min(nums1[0]*nums2[0], nums1[0]*nums2[-1], nums1[-1]*nums2[0], nums1[-1]*nums2[-1])
    high = max(nums1[0]*nums2[0], nums1[0]*nums2[-1], nums1[-1]*nums2[0], nums1[-1]*nums2[-1])

    #    k
    while low <= high:
        mid = (low + high) // 2
        count = countPairs(nums1, nums2, mid)

        if count < k:
            low = mid + 1
        else:
            high = mid - 1

    return low

def countPairs(nums1, nums2, target):
    #     <= target
    count = 0
    #  nums1     nums2
    for num1 in nums1:
        if num1 > 0:
            #   nums2  <= target/num1
            threshold = target / num1
            #  nums2    > threshold
            left, right = 0, len(nums2)
            while left < right:
                mid = (left + right) // 2
                if nums2[mid] > threshold:
                    right = mid
                else:
                    left = mid + 1
            count += left
        elif num1 < 0:
            #   nums2  >= target/num1
            threshold = target / num1
            #  nums2    >= threshold
            left, right = 0, len(nums2)
            while left < right:
                mid = (left + right) // 2
                if nums2[mid] >= threshold:
                    right = mid
                else:
                    left = mid + 1
            count += len(nums2) - left
        else: # num1 == 0
            if target >= 0:
                count += len(nums2)
    return count
```

Second Array



Variables

low	-5
high	-3
mid	-4
count	4
current_num1	3
threshold	-1.33

3: nums2 -1.33