

ARRAY : Video - 60

Hard
Leetcode
- 4

E.A.S.Y...



Median Of Two Sorted Arrays (PART-1)

Brute
Better

Facebook
Instagram } → code story with MIK

(Twitter) → CS with MIK

Company :- Google

Microsoft



4. Median of Two Sorted Arrays

Hard 25819 2843 Add to List Share

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 $O(\log(m+n))$.

(Pen+2)

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 $(2 + 3) / 2 = 2.5$.

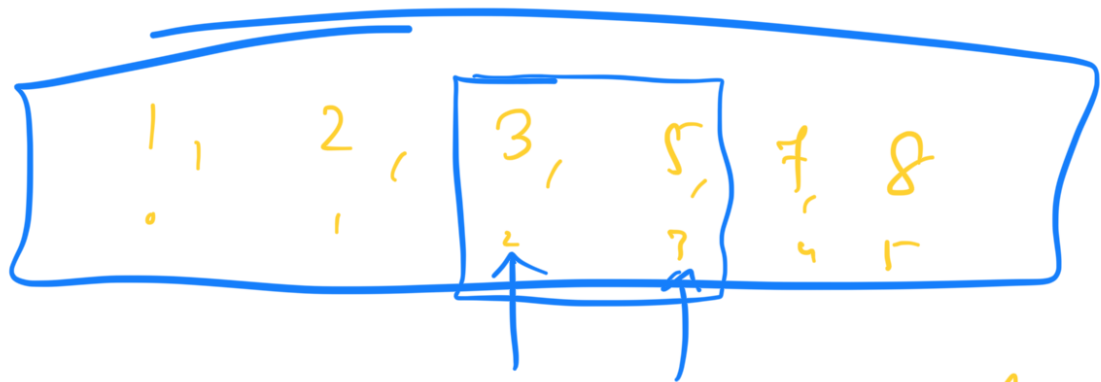
1, 2
3, 4

$(2+3)/2$

$4/2 = 2$
 $4/2 - 1 =$

Median :- (middle number)





$$(3 + 5) / 2 = 4$$

$$n = 6$$

$$n/2 = 3$$

$$(n/2 - 1) = 2$$

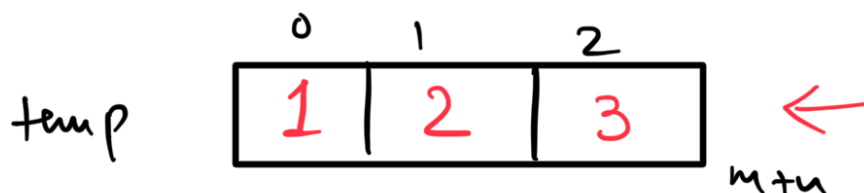
Brute Force :-

$$\text{nums1} = [1, 3]^i \rightarrow m$$

$$\text{nums2} = [2]^j \rightarrow n$$

$$\text{nums1}[i]$$

$$\text{nums2}[j]$$

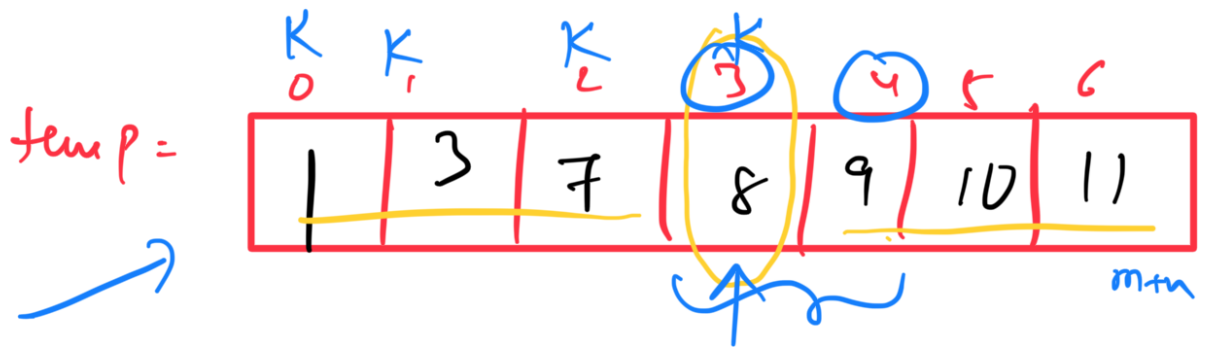


space
 $O(m+n)$
 \vdots
 \cap

$$\text{nums1} = [1, 3, 7]^i$$

$$j \quad j \quad j \quad j \quad j$$

nums2 = [8, 9, 10, 11]



size = mid = 7

$7/2 = 3$

if (size / 2 == 1)

return temp[size/2];

8

return (temp[size/2] + temp[size/2 - 1]) / 2.0;

Better Approach:-

O(1) space.

$$\text{nums1} = \{1, 3, 7\}, m = 3$$

$$\text{nums2} = \{8, 9, 10, 11, 12\}, n = 5$$

$\text{size} = (m+n) = 8$
 $\text{idx1} = \frac{3}{2} = 3$ (size/2 - 1)
 $\text{idx2} = \frac{4}{2} = 4$ (size/2)

element1 = 8
 element2 = 9

$\lceil \text{size} / 2 \rceil$

$$K = \cancel{5} \neq 8$$

if (size/2 == 1) { // odd
 return element2;
 }

return (element1 + element2) / 2.0;

func(K, ...)

~~func(K, ...)~~

Approach-1

```
if (nums1[i] < nums2[j]) {  
    temp[k++] = nums1[i++];  
} else {  
    temp[k++] = nums2[j++];  
}
```

```
while (i < m) {  
    temp[k++] = nums1[i++];  
}
```

```
while (j < n)  
    temp[k++] = nums2[j++];  
}
```

Approach-2

```
if (nums1[i] < nums2[j]) {  
    if (k == idx1)  
        elem1 = nums1[i];  
  
    if (k == idx2)  
        elem2 = nums1[i];  
    i++;  
}
```

```
else {  
    if (k == idx1)  
        elem1 = nums2[j];  
  
    if (k == idx2)  
        elem2 = nums2[j];  
    j++;  
}
```

k++;

}

```
while (i < m) {  
    if (k == idx1)  
        elem1 = nums1[i]  
    if (k == idx2)  
        elem2 = nums1[i]  
    i++;  
    k++;  
}
```

```
while (j < n) {
```

num2[j];

}
