Median of Two Sorted Arrays

Try to solve the Median of Two Sorted Arrays problem.

We'll cover the following

Statement

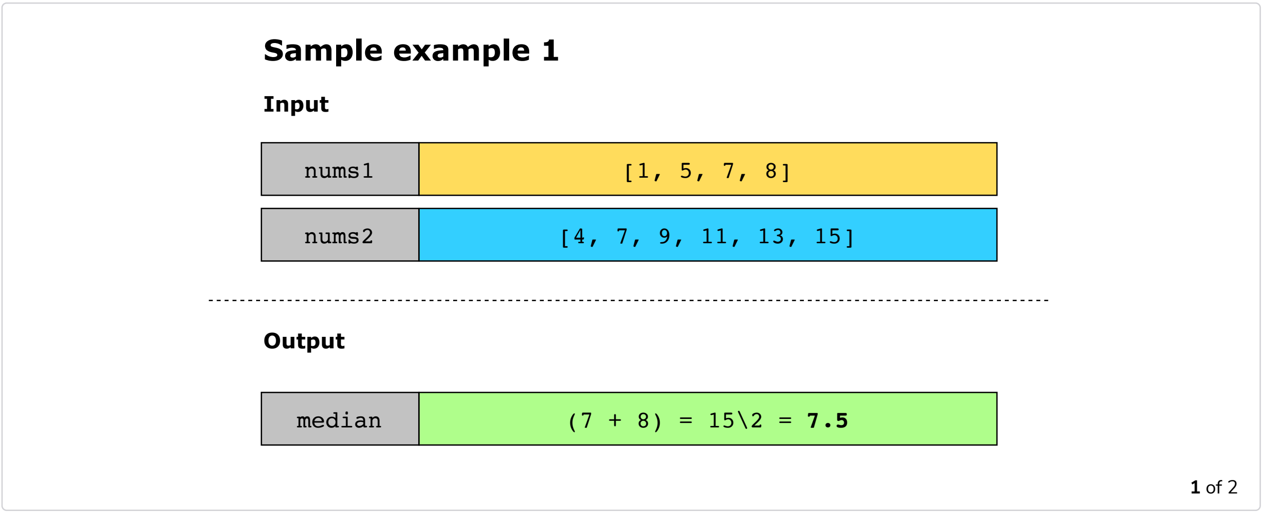
You're given two sorted integer arrays, `nums1` and `nums2`, of size m and n, respectively. Your task is to return the median of the two sorted arrays.

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

Constraints:

- `nums1.length` $== m$
- `nums2.length` $== n$
- $0 \leq m \leq 1000$
- $0 \leq n \leq 1000$
- $1 \leq m + n \leq 2000$
- $-10^6 \leq$ `nums1[i]`, `nums2[i]` $\leq 10^6$

Examples



Understand the problem

Let’s take a moment to make sure you’ve correctly understood the problem. The quiz below helps you check if you’re solving the correct problem:

Median of Two Sorted Arrays

1 What is the output if the following arrays are given as input?

nums1 = [3, 5, 9, 10, 14]

nums2 = [1, 4, 6, 9, 30]

A) 7.0

B) 8.0

C) 7.5

Submit Answer

Question 1 of 2
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

> **Note:** As an additional challenge, we have intentionally hidden the solution to this puzzle.

Drag and drop the cards to rearrange them in the correct sequence.

Otherwise, if **left short** is greater than **right long**, move **left short** to halfway between its current position and the start of the shorter array, then figure out **left long** and update **right short** and **right long** accordingly.

Repeat the Check Partition step and either return the median or move the partition.

Otherwise, if **left long** is greater than **right short**, move **left long** to halfway between its current position and the start of the longer array, and update **right long**, **left short** and **right short** accordingly.

Check Partition: If **left long** is less than **right short** and **left short** is less than **right long**, we can calculate the median of the two sorted arrays as the mean of **Max(left long, left short)** and **Min(right long, right short)**.

Pick the middle element of the longer array as the partition location. Let's call this element **left long**. Let's call the very next element **right long**.

Figure out how many elements of the shorter array need to be in the first half. Let's call this element **left short**. Let's call the next element **right short**.

Reset Submit

Try it yourself

Implement your solution in the following coding playground.

> **Note:** We have left the solution to this challenge as an exercise for you. You may try to translate the logic of the solved puzzle into a coded solution.

```
public class FindMedian{
   public static float findMedian(int[] nums1, int[] nums2) {

      // Your code will replace this placeholder return statement

      return 0;
```

```
    }
}
```

Powered by AI

Submit

Test Cases Results

Case 1 Case 2 Case 3

Input #1

[1,2]

Input #2

[3,4]

Median of Two Sorted Arrays

← Back

Solution: Kth Smallest...

Next →

Top K Elements: Introd...

✓ Mark as Completed