

```

import java.util.PriorityQueue;

public class new_approach_OfMedian {

    public static int median(int[] ans) {
        int n = ans.length;
        if (n % 2 == 0) {
            int z = n / 2;
            int e = ans[z];
            int q = ans[z - 1];

            return (e + q) / 2;
        } else {
            int z = Math.round(n / 2);
            return ans[z];
        }
    }

    public static void heap_sort(int[] ans) {
        PriorityQueue<Integer> min_heap = new PriorityQueue<>();
        for (int idx : ans) {
            min_heap.add(idx);
        }
        for (int i = 0; min_heap.size() > 0; i++, min_heap.poll()) {
            ans[i] = min_heap.peek();
        }
    }

    public static void main(String[] args) {
        int[] arr1 = { -5, 3, 6, 12, 15 };
        int[] arr2 = { -12, -10, -6, -3, 4, 10 };

        int i = arr1.length;
        int j = arr2.length;

        int[] ans = new int[i + j];
        System.arraycopy(arr1, 0, ans, 0, i);
        System.arraycopy(arr2, 0, ans, i, j);

        heap_sort(ans);
        System.out.println(median(ans));
    }
}

```