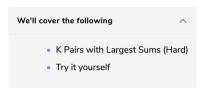




# Problem Challenge 1



### K Pairs with Largest Sums (Hard) #

Given two sorted arrays in descending order, find 'K' pairs with the largest sum where each pair consists of numbers from both the arrays.

#### Example 1:

```
Input: L1=[9, 8, 2], L2=[6, 3, 1], K=3
Output: [9, 3], [9, 6], [8, 6]
Explanation: These 3 pairs have the largest sum. No other pair has a sum larger than any of the se.
```

#### Example 2:

```
Input: L1=[5, 2, 1], L2=[2, -1], K=3
Output: [5, 2], [5, -1], [2, 2]
```

## Try it yourself #

Try solving this question here:

```
🚣 Java
               Python3
                                           G C++
       class LargestPairs {
         public static List<int[]> findKLargestPairs(int[] nums1, int[] nums2, int k) {
           List<int[]> result = new ArrayList<>();
         public static void main(String[] args) {
           int[] l1 = new int[] { 9, 8, 2 };
int[] l2 = new int[] { 6, 3, 1 };
           List<int[]> result = LargestPairs.findKLargestPairs(l1, l2, 3);
            System.out.print("Pairs with largest sum are: ");
for (int[] pair : result)
              System.out.print("[" + pair[0] + ", " + pair[1] + "] ");
   Run
                                                                                                     Save Reset []
                                                                                                                   Next →
 ← Back
Smallest Number Range (Hard)
                                                                                             Solution Review: Problem Challenge 1
                                                                                                         ✓ Mark as Completed
```