Our Solution(s) Run Code

Solution 1

```swift
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

class Program {
    func smallestDifference(arrayOne: inout [Int], arrayTwo: inout [Int]) -> [Int]
        arrayOne.sort()
        arrayTwo.sort()

        var idxOne = 0
        var idxTwo = 0

        var current = Int.max
        var smallest = Int.max

        var smallestPair: [Int] = []

        while idxOne < arrayOne.count, idxTwo < arrayTwo.count {
            let firstNum = arrayOne[idxOne]
            let secondNum = arrayTwo[idxTwo]

            if firstNum < secondNum {
                current = secondNum - firstNum
                idxOne = idxOne + 1
            } else if firstNum > secondNum {
                current = firstNum - secondNum
                idxTwo = idxTwo + 1
            } else {
                return [firstNum, secondNum]
            }

            if smallest > current {
                smallest = current
                smallestPair = [firstNum, secondNum]
            }
        }

        return smallestPair
    }
}

```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```swift
class Program {
    func smallestDifference(arrayOne: inout [Int], arrayTwo: inout [Int]) -> [Int]
        // Write your code here.
        return []
    }
}

```

Custom Output Raw Output Submit Code

Run or submit code when you're ready.