Our Solution(s) Run Code

Solution 1

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(nlog(n) + mlog(m)) time | O(1) space
def smallestDifference(arrayOne, arrayTwo):
    arrayOne.sort()
    arrayTwo.sort()
    idxOne = 0
    idxTwo = 0
    smallest = float("inf")
    current = float("inf")
    smallestPair = []
    while idxOne < len(arrayOne) and idxTwo < len(arrayTwo):
        firstNum = arrayOne[idxOne]
        secondNum = arrayTwo[idxTwo]
        if firstNum < secondNum:
            current = secondNum - firstNum
            idxOne += 1
        elif secondNum < firstNum:
            current = firstNum - secondNum
            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

```python
def smallestDifference(arrayOne, arrayTwo):
    # Write your code here.
    pass

```

Custom Output Raw Output Submit Code

Run or submit code when you're ready.