

Our Solution(s)

Run Code

Solution 1

```
1 # Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 # O(nlog(n) + mlog(m)) time | O(1) space
4 def smallestDifference(arrayOne, arrayTwo):
5     arrayOne.sort()
6     arrayTwo.sort()
7     idxOne = 0
8     idxTwo = 0
9     smallest = float("inf")
10    current = float("inf")
11    smallestPair = []
12    while idxOne < len(arrayOne) and idxTwo < len(arrayTwo):
13        firstNum = arrayOne[idxOne]
14        secondNum = arrayTwo[idxTwo]
15        if firstNum < secondNum:
16            current = secondNum - firstNum
17            idxOne += 1
18        elif secondNum < firstNum:
19            current = firstNum - secondNum
20            idxTwo += 1
21        else:
22            return [firstNum, secondNum]
23        if smallest > current:
24            smallest = current
25            smallestPair = [firstNum, secondNum]
26    return smallestPair
27
```

Your Solutions

Run Code

Solution 1   Solution 2   Solution 3

```
1 def smallestDifference(arrayOne, arrayTwo):
2     # Write your code here.
3     pass
4
```

Our Tests

Custom Output

Submit Code

