

Our Solution(s)	Run Code	Your Solutions	Run Code
-----------------	----------	----------------	----------

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 #include <vector>
4 #include <algorithm>
5 #include <climits>
6 using namespace std;
7
8 // O(nlog(n) + mlog(m)) time | O(1) space
9 vector<int> smallestDifference(vector<int> arrayOne, vector<int>
10     sort(arrayOne.begin(), arrayOne.end());
11     sort(arrayTwo.begin(), arrayTwo.end());
12     int idxOne = 0;
13     int idxTwo = 0;
14     int smallest = INT_MAX;
15     int current = INT_MAX;
16     vector<int> smallestPair;
17     while (idxOne < arrayOne.size() && idxTwo < arrayTwo.size()) {
18         int firstNum = arrayOne[idxOne];
19         int secondNum = arrayTwo[idxTwo];
20         if (firstNum < secondNum) {
21             current = secondNum - firstNum;
22             idxOne++;
23         } else if (secondNum < firstNum) {
24             current = firstNum - secondNum;
25             idxTwo++;
26         } else {
27             return vector<int>{firstNum, secondNum};
28         }
29         if (smallest > current) {
30             smallest = current;
31             smallestPair = {firstNum, secondNum};
32         }
33     }
```

Solution 1Solution 2Solution 3

```
1 #include <vector>
2 using namespace std;
3
4 vector<int> smallestDifference(vector<int> arrayOne, vector<int>
5     // Write your code here.
6     return {};
7 }
8
```

```
1 #!/usr/bin/env python3
2
3 """
4     Test suite for the 'is_prime' function.
5     This script tests the 'is_prime' function against a range of
6     prime and composite numbers.
7 """
8
9 import sys
10
11 def is_prime(n):
12     """
13     Check if a number is prime.
14     Returns True if n is prime, False otherwise.
15     """
16     if n < 2:
17         return False
18     for i in range(2, int(n**0.5) + 1):
19         if n % i == 0:
20             return False
21     return True
22
23 # Test cases
24 test_cases = [
25     2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47,
26     53, 59, 61, 67, 71, 73, 79, 83, 89, 97,
27     101, 103, 107, 109, 113, 127, 131, 137, 139, 149,
28     151, 157, 163, 167, 173, 179, 181, 187, 191, 193,
29     197, 199, 211, 223, 227, 229, 233, 239, 241, 251,
30     257, 263, 269, 271, 277, 281, 283, 293, 307, 311,
31     313, 317, 331, 337, 347, 349, 353, 359, 367, 373,
32     379, 383, 389, 397, 401, 409, 419, 421, 431, 433,
33     439, 443, 449, 457, 461, 463, 467, 479, 487, 491,
34     499, 503, 509, 521, 523, 527, 539, 541, 547, 557,
35     563, 569, 571, 577, 587, 593, 599, 601, 607, 613,
36     617, 619, 631, 637, 641, 643, 647, 653, 659, 661,
37     673, 677, 683, 687, 691, 697, 701, 709, 713, 719,
38     727, 733, 739, 743, 751, 757, 761, 769, 773, 787,
39     797, 809, 811, 821, 823, 827, 829, 833, 839, 847,
40     853, 857, 859, 863, 877, 881, 883, 887, 893, 899,
41     907, 911, 919, 929, 937, 941, 947, 953, 967, 971,
42     977, 983, 989, 991, 997
43 ]
44
45 # Run tests
46 for n in test_cases:
47     if is_prime(n):
48         print(f"{n} is prime")
49     else:
50         print(f"{n} is not prime")
51
52 # End of script
```

Run or submit code when you're ready.