Our Solution(s) Run Code

Solution 1

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

class Program {
  // O(n) time | O(1) space
  public static int kadanesAlgorithm(int[] array) {
    int maxEndingHere = array[0];
    int maxSoFar = array[0];
    for (int i = 1; i < array.length; i++) {
      int num = array[i];
      maxEndingHere = Math.max(num, maxEndingHere + num);
      maxSoFar = Math.max(maxSoFar, maxEndingHere);
    }
    return maxSoFar;
  }
}

```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```java
class Program {
  public static int kadanesAlgorithm(int[] array) {
    // Write your code here.
    return -1;
  }
}

```

Our Tests

Custom Output Submit Code

```
@Test
public void Test0(){
    int[] input = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    Utils.assertTrue(Program.maximumSum(input) == 55);
}

@Test
public void Test1(){
    int[] input = {-1, -2, -3, -4, -5, -6, -7, -8, -9, -10};
    Utils.assertTrue(Program.maximumSum(input) == -1);
}

@Test
public void Test2(){
    int[] input = {100, -2, -3, -4, -5, -6, -7, -8, -9, -10};
    Utils.assertTrue(Program.maximumSum(input) == 100);
}

@Test
public void Test3(){
    int[] input = {1, 2, 3, 4, 5, 6, -20, 7, 8, 9, 10};
```

Run or submit code when you're ready.