

Our Solution(s)

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 class Program {
4     // O(n) time | O(1) space - where n is the length of the input array
5     func longestPeak(array: [Int]) -> Int {
6         var longestPeakLength = 0
7         var i = 1
8         while i < array.count - 1 {
9             let isPeak = array[i - 1] < array[i] && array[i] > array[i + 1]
10            if !isPeak {
11                i += 1
12                continue
13            }
14
15            var leftIdx = i - 2
16            while leftIdx >= 0, array[leftIdx] < array[leftIdx + 1] {
17                leftIdx -= 1
18            }
19
20            var rightIdx = i + 2
21            while rightIdx < array.count, array[rightIdx] < array[rightIdx - 1] {
22                rightIdx += 1
23            }
24
25            let currentPeakLength = rightIdx - leftIdx - 1
26            if currentPeakLength > longestPeakLength {
27                longestPeakLength = currentPeakLength
28            }
29            i = rightIdx
30        }
31        return longestPeakLength
32    }
33 }
```

Your Solutions

Run Code

Solution 1

Solution 2

Solution 3

```
1 class Program {
2     func longestPeak(array: [Int]) -> Int {
3         // Write your code here.
4         return -1
5     }
6 }
7
```

Custom Output

Submit Code

```

1 def test1():
2     program = Program()
3     testenv("Test Case 1") & (0) success = test 20
4     array = [20, 10]
5     expected = 5
6     actual = program.longestSub(array, array)
7     assertEquals(expected, actual)
8 }
9
10 testenv("Test Case 2") & (0) success = test 20
11 array = [20, 10, 20]
12 expected = 5
13 actual = program.longestSub(array, array)
14 assertEquals(expected, actual)
15 }
16
17 testenv("Test Case 3") & (0) success = test 20
18 array = [20, 10, 10, 10, 10, 20]
19 expected = 5
20 actual = program.longestSub(array, array)
21 assertEquals(expected, actual)
22 }
23
24 testenv("Test Case 4") & (0) success = test 20

```

Run or submit code when you're ready.