

Our Solution(s)		Run Code	Your Solutions			Run Code
Solution 1			Solution 1	Solution 2	Solution 3	
<pre>1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved. 2 3 package main 4 5 // O(n) time O(1) space - where n is the length of the input : 6 func HasSingleCycle(array []int) bool { 7 numElementsVisited := 0 8 currentIdx := 0 9 for numElementsVisited < len(array) { 10 if numElementsVisited > 0 && currentIdx == 0 { 11 return false 12 } 13 numElementsVisited++ 14 currentIdx = getNextIdx(currentIdx, array) 15 } 16 return currentIdx == 0 17 } 18 19 func getNextIdx(currentIdx int, array []int) int { 20 jump := array[currentIdx] 21 nextIdx := (currentIdx + jump) % len(array) 22 if nextIdx >= 0 { 23 return nextIdx 24 } 25 return nextIdx + len(array) 26 } 27</pre>			<pre>1 package main 2 3 func HasSingleCycle(array []int) bool { 4 // Write your code here. 5 return false 6 } 7</pre>			
Our Tests			Custom Output			Submit Code

