

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 type LinkedList struct {
6     Value int
7     Next *LinkedList
8 }
9
10 // O(n) time | O(1) space - where n is the number of nodes in the Link
11 func ReverseLinkedList(head *LinkedList) *LinkedList {
12     var p1, p2 *LinkedList = nil, head
13     for p2 != nil {
14         p3 := p2.Next
15         p2.Next = p1
16         p1 = p2
17         p2 = p3
18     }
19     return p1
20 }
21
```

Solution 1   Solution 2   Solution 3

```
1 package main
2
3 type LinkedList struct {
4     Value int
5     Next *LinkedList
6 }
7
8 func ReverseLinkedList(head *LinkedList) *LinkedList {
9     // Write your code here.
10    return nil
11 }
12
```

Our Tests



Custom Output

Submit Code

```

24. 1. WordCountCollector (10%)
25. wordCount = 0
26. for word in input.split():
27.     wordCount += len(word)
28. wordCount = wordCount
29. return wordCount
30. def main():
31. wordCount = WordCountCollector(input)
32. print wordCount
33. if __name__ == '__main__':
34.     main()
35. wordCount = WordCountCollector(input)
36. print wordCount
37. wordCount = wordCount

```

Run or submit code when you're ready.