Quiz 3

Due Jul 7, 2021 at 11:59pm **Points** 100 **Questions** 2
Available Jul 6, 2021 at 11:59pm - Jul 7, 2021 at 11:59pm 24 hours
Time Limit 30 Minutes

Instructions

Dear all,

The third quiz of this course is released today. This quiz will be available on Canvas from **11:59 pm July 6 to 11:59 pm July 7** (24 hours, **U.S. Central Time**) so you can take it anytime during this period. Moreover, you have **30 minutes** with only **ONE attempt** to complete and turn in your answers so please make sure you complete all questions before submitting.

NOTE: Your fill-in-the-blank answers shown when you finish the exam may not match our pre-defined answers and we will revise them manually so please do not worry about it.

Good luck!

Your TAs

This quiz is no longer available as the course has been concluded.

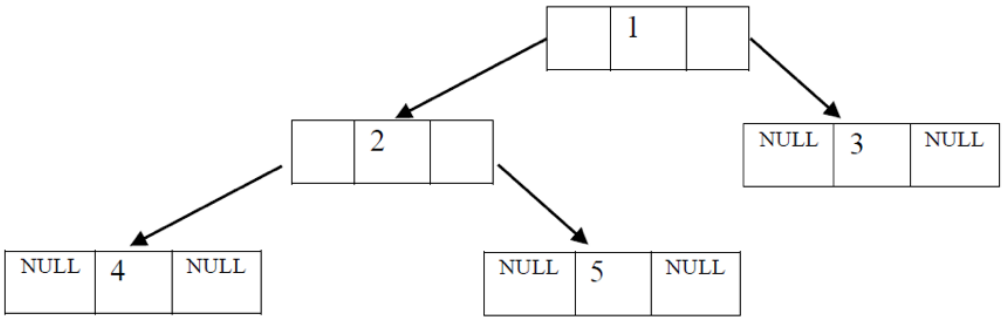
Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	29 minutes	100 out of 100

ⓘ Correct answers are no longer available.

Score for this quiz: **100** out of 100
Submitted Jul 7, 2021 at 6:15pm
This attempt took 29 minutes.

Question 1	**40 / 40 pts**



Please fill in the blank lines to define each node from a transformed singly linked list. You can use the notations "left" and "right".

```
1. struct Node
2. {
3.     int data;
4.     struct [Node* left] ;
5.     struct [Node* right] ;
6.
7.     Node(int data)
8.     {
9.         this->data = data;
10.        [left = NULL] ;
11.        [right = NULL] ;
12.    }
13. };
```

Answer 1:

Node* left

Answer 2:

Node* right

Answer 3:

left = NULL

Answer 4:

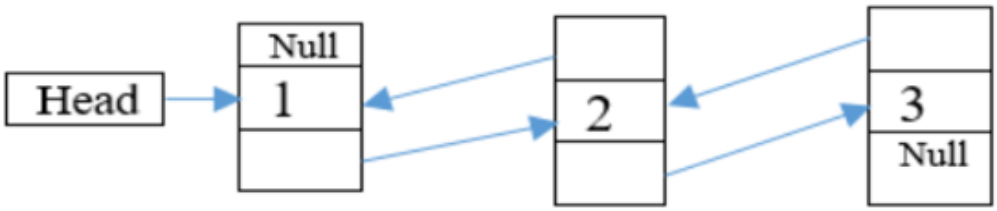
right = NULL

Partial

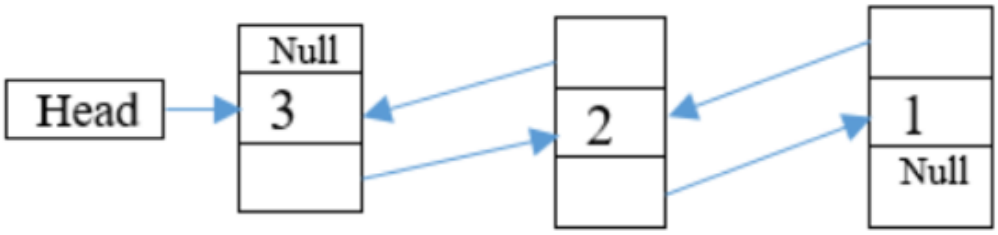
Question 2

60 / 60 pts

Before



After



Please fill in the blank lines to reverse a doubly linked list with a recursion.

```
1. Node* Reverse(Node* node)
2. {
3.     // If empty list, return
4.     if (!node)
5.         return [NULL] ;
6.
7.     // swap
8.     Node* temp = [node-next] ;
9.     node->next = [node-prev] ;
10.    node->prev = [temp] ;
11.
12.    // If the prev is now NULL, the list
```

```
13.     // has been fully reversed
14.     if ( !node-prev )
15.         return node;
16.
17.     // Otherwise, keep going
18.     return Reverse( node-prev );
19. }
```

Answer 1:

NULL

Answer 2:

node-next

Answer 3:

node-prev

Answer 4:

temp

Answer 5:

!node-prev

Answer 6:

node-prev

Quiz Score: **100** out of 100