

**DAILY ONLINE ACTIVITIES SUMMARY**

<b>Date:</b>	<b>20-5-2020</b>	<b>Name:</b>	<b>Nagashree D</b>
<b>Sem &amp; Sec</b>	<b>8<sup>th</sup> ,A</b>	<b>USN:</b>	<b>4AL16CS055</b>
<b>Online Test Summary</b>			
<b>Subject</b>	<b>IOT(IAI)</b>		
<b>Max. Marks</b>	<b>30</b>	<b>Score</b>	<b>19</b>
<b>Certification Course Summary</b>			
<b>Course</b>	<b>Introduction to ethical hacking</b>		
<b>Certificate Provider</b>	<b>Great Learner Academy</b>	<b>Duration</b>	<b>6Hrs</b>
<b>Coding Challenges</b>			
<b>Problem Statement:</b> Write a C Program to Reverse a Linked List in groups of given size			
<b>Status:</b> Solved			
<b>Uploaded the report in Github</b>		<b>Yes</b>	
<b>If yes Repository name</b>		<b>Nagashreed</b>	
<b>Uploaded the report in slack</b>		<b>Yes</b>	

# Online test Details

## Test Completed!

You have successfully participated in IOT IA1.

**Rate this Test**

Your Rating: ★★★★★ ◀ Click to Rate

Results

Analytics

✓ MCQ


Your Score **19** / 30

# Certification Course Details:

**greatlearning**  
Learning for Life

HomeLive Sessions

My

Introduction to Ethical Hacking

Course In Progress

CONTENTASSESSMENTS

Learning Videos

	Career and Growth Ladder in Ethical Hacking	18m	
	Domains and Process Implementation under Ethical Hacking	54m	
	Ethical Hacking in Network Architecture-Demonstration	48m	
	Ethical Hacking in Web Applications-Demonstration	50m	
	Ethical Hacking on Mobile Platforms-Demonstration	34m	

## Domains and Process Implementation under Ethical Hacking

- Web Application Domain
- Mobile
- Network Architecture Domain

## Hacking Methodology

- Web Footprinting – Gathering Information
- Vulnerability Scanners – w3af, Acunetix
- Identity Entry Points and Attack Surface

## Coding Challenges:

Write a C Program to Reverse a Linked List in groups of given size

Test Case 1:

If a linked list is:  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$

The value of size k is 2

Then the linked list looks like:  $2 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow 6 \rightarrow 5 \rightarrow 8 \rightarrow 7$

Test Case 2:

If a linked list is:  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$

The value of size k is 3

Then the linked list looks like:  $3 \rightarrow 2 \rightarrow 1 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow 8 \rightarrow 7$

Code:

```
#include<stdio.h>
#include<stdlib.h>
struct Node
{
    int data;
    struct Node* next;
};
```

```
pointer to the new head node. /
struct Node reverse (struct Node head, int k)
{
    struct Node current = head;
    struct Node next = NULL;
    struct Node prev = NULL;
    int count = 0;
```

```
while (current != NULL && count < k)
{
    next = current->next;
    current->next = prev;
    prev = current;
    current = next;
    count++;
}
```

```
if (next != NULL)
    head->next = reverse(next, k);
```

```
return prev;
}

void push(struct Node** head_ref, int new_data)
{
    struct Node* new_node =
    (struct Node*) malloc(sizeof(struct Node));
```

```
new_node->data = new_data;

new_node->next = (*head_ref);

(*head_ref) = new_node;
}
```

```
void printList(struct Node *node)
{
    while (node != NULL)
    {
        printf("%d ", node->data);
        node = node->next;
    }
}
```

```
int main(void)
{
    struct Node* head = NULL;
    push(&head, 8);
    push(&head, 7);
    push(&head, 6);
    push(&head, 5);
    push(&head, 4);
    push(&head, 3);
    push(&head, 2);
    push(&head, 1);
```

```
    printf("\nGiven linked list \n");
    printList(head);
    head = reverse(head, 2);

    printf("\nReversed Linked list \n");
    printList(head);

    return(0);
}
```