DAILY ONLINE ACTIVITIES SUMMARY

Date:	21/05/2020	**Name:**	Shaima Abdul Kader
Sem & Sec	8 ,B sec	**USN:**	4A16CS087

Online Test Summary

Subject	SMS 2		
Max. Marks	60	**Score**	28

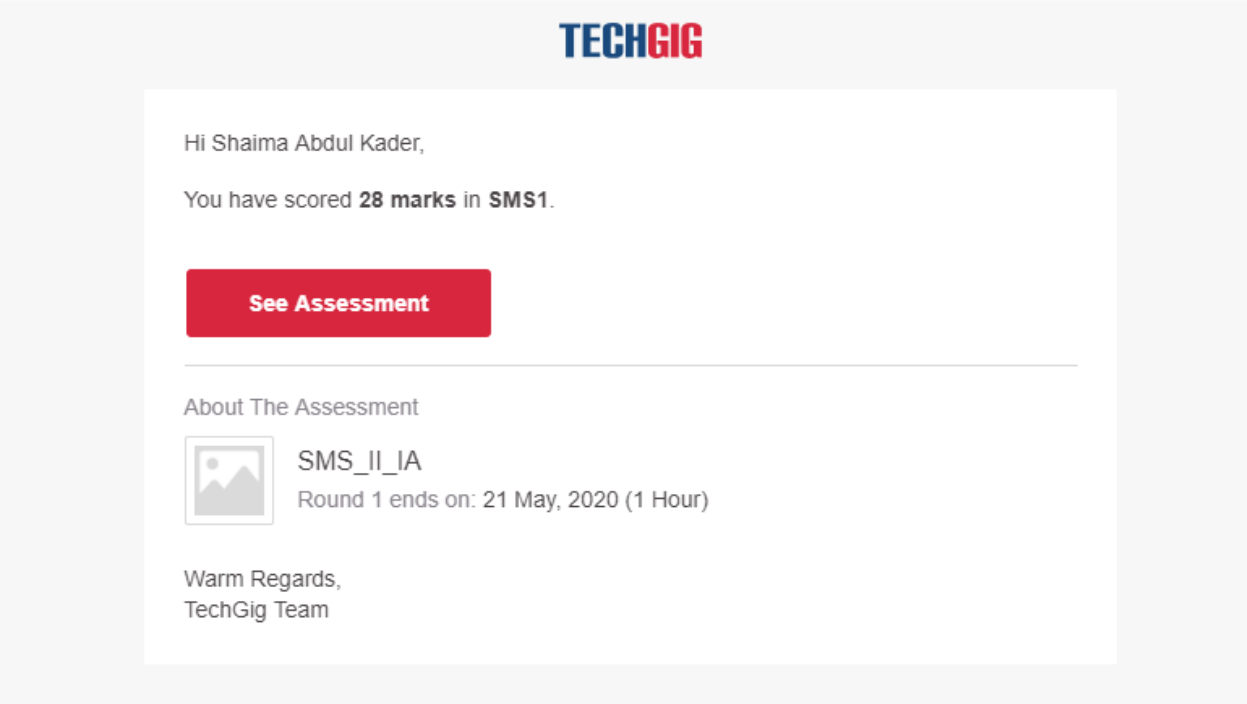
Certification Course Summary

Course	Cybersecurity Essentials		
Certificate Provider	ICT ACADEMY	**Duration**	5 hrs.

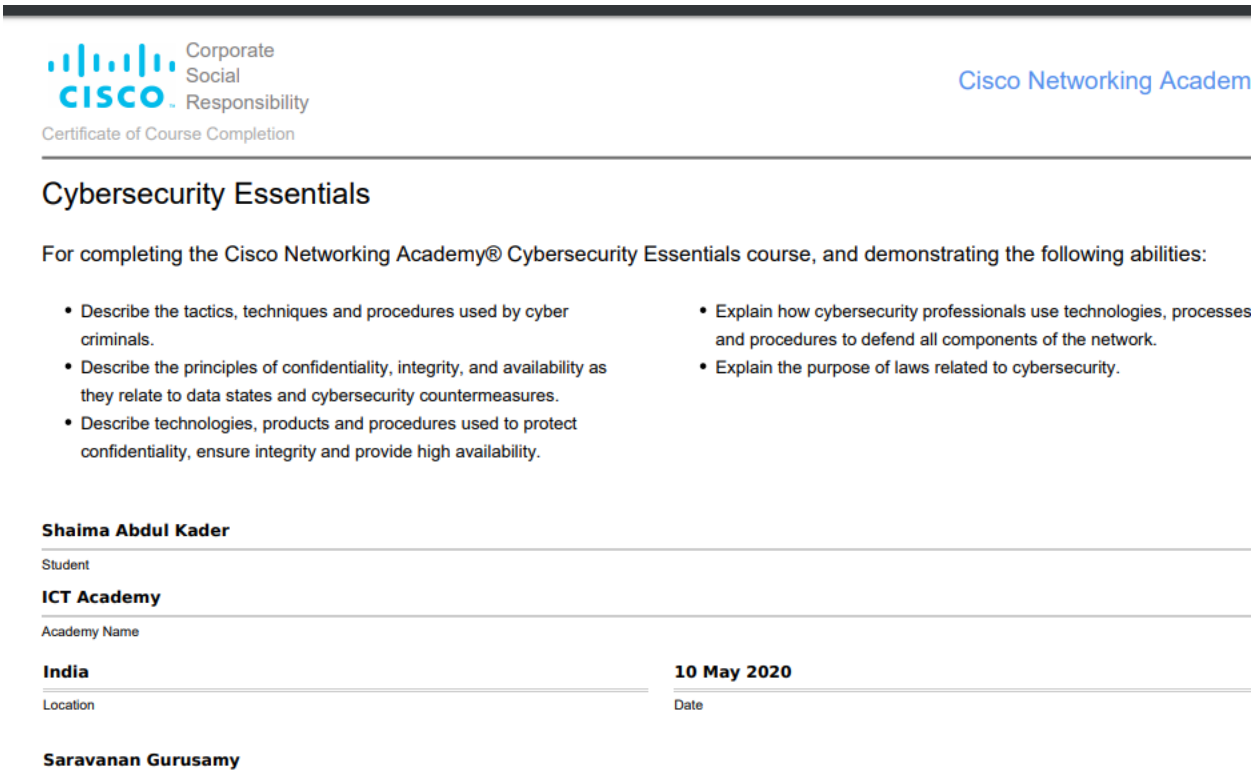
Coding Challenges

Problem Statement: C Program to Reverse a Linked List in groups of given size.	
Status: Completed.	
Uploaded the report in Github	Yes
If yes Repository name	shaima
Uploaded the report in slack	Yes.

Online Test Details: (Attach the snapshot and briefly write the report for the same)



Certification Course Details: (Attach the snapshot and briefly write the report for the same)



Corporate Social Responsibility

Cisco Networking Academ

Certificate of Course Completion

Cybersecurity Essentials

For completing the Cisco Networking Academy® Cybersecurity Essentials course, and demonstrating the following abilities:

- Describe the tactics, techniques and procedures used by cyber criminals.
- Describe the principles of confidentiality, integrity, and availability as they relate to data states and cybersecurity countermeasures.
- Describe technologies, products and procedures used to protect confidentiality, ensure integrity and provide high availability.
- Explain how cybersecurity professionals use technologies, processes and procedures to defend all components of the network.
- Explain the purpose of laws related to cybersecurity.

Shaima Abdul Kader

Student

ICT Academy

Academy Name

India

Location

10 May 2020

Date

Saravanan Gurusamy

Coding Challenges Details:

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

```
#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));
```

.....