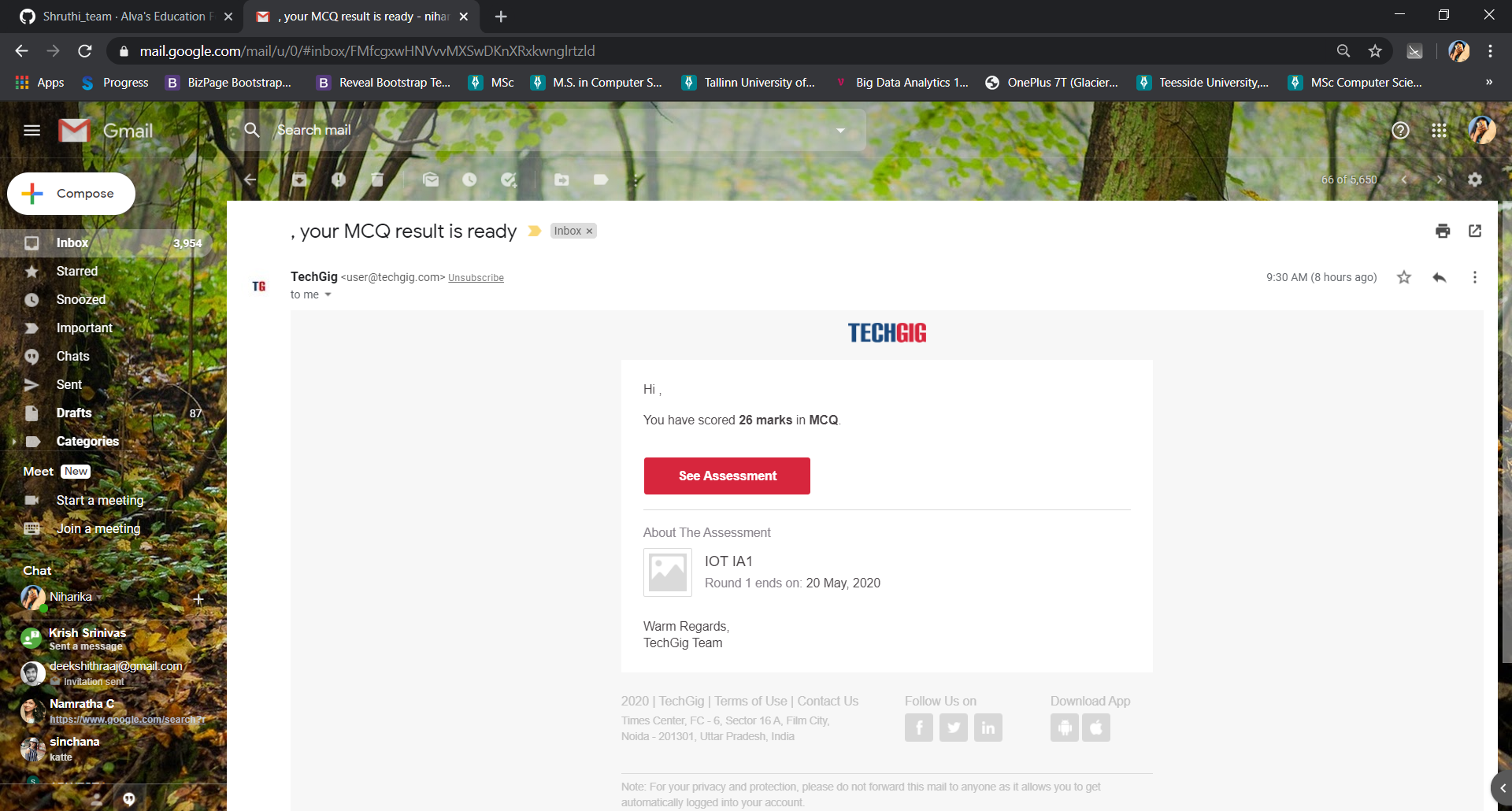
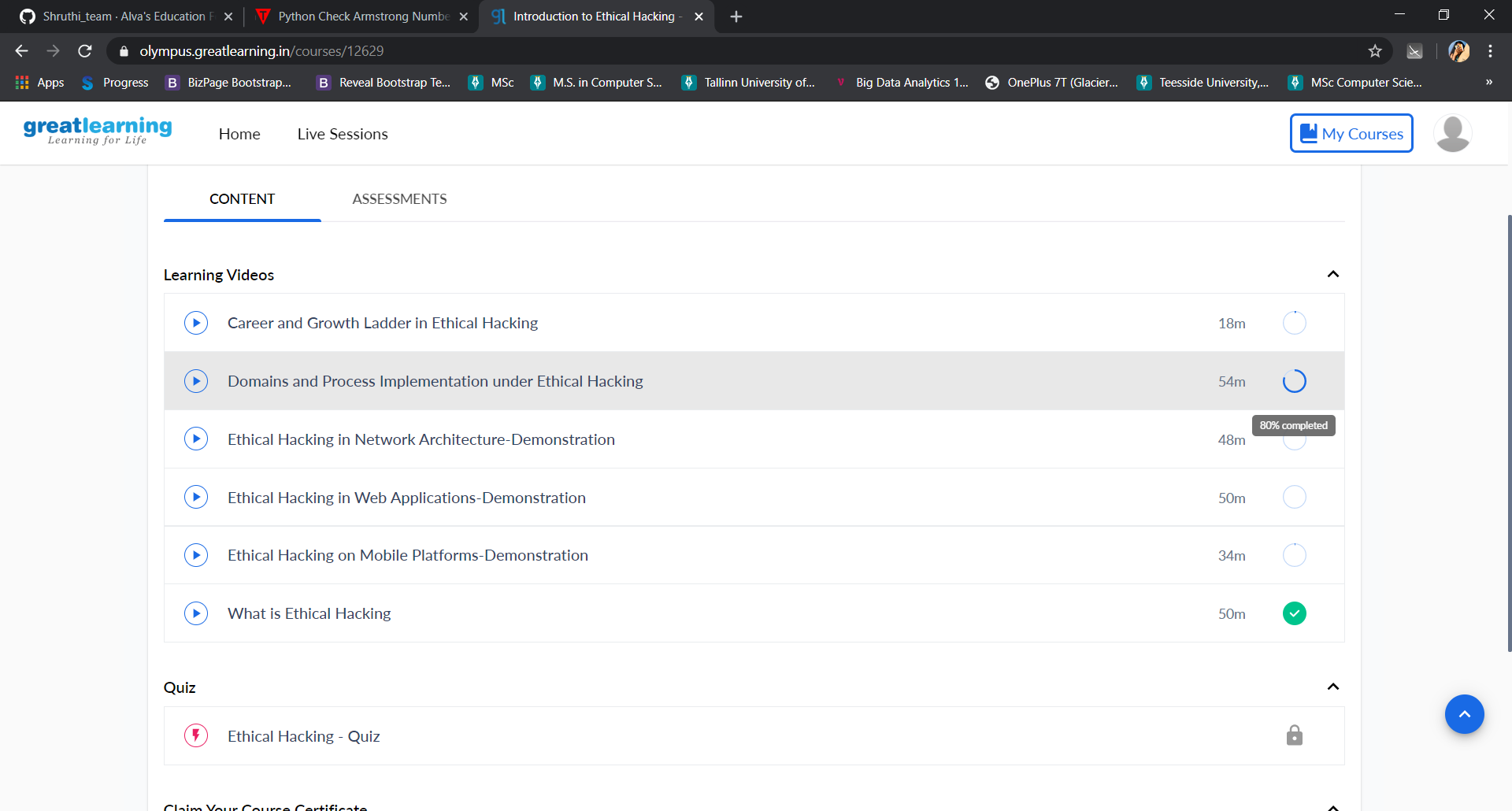
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **20-05-2020** | | | | | **Name:** | **Niharika G V** | |
| **Sem & Sec** | **8 A** | | | | | **USN:** | **4AL16CS059** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **IOT** | | | | | | |
| **Max. Marks** | | **30** | | **Score** | | | **26** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Introduction to Ethical Hacking** | | | | | | | |
| **Certificate Provider** | | | **Bright Learning** | | **Certificate Provider** | | | **Bright Learning** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**   1. **Genrating Armstrong numbers using Python** 2. **Write a C Program to Reverse a Linked List in groups of given size.**   **Test Case 1:**  **If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8**  **The value of size k is 2**  **Then the linked list looks like: 2 → 1 → 4 → 3 → 6 → 5 → 8 → 7**  **Test Case 2:**  **If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8**  **The value of size k is 3**  **Then the linked list looks like: 3 → 2 → 1 → 6 → 5 → 4 → 8 → 7** | | | | | | | | |
| **Status: Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Yes** | | | |
| **If yes Repository name** | | | | | **Daily Progress Reports** | | | |
| **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)



In today’s topic, the Domain and process Implementation under Ethical hacking have been discussed.

The main domains that were discussed are

1. Web Application Domain
2. Mobile
3. Network Architecture Domain

After which the 3 main methodologies in hacking were discussed

1. Web footprinting – Gathering information
2. Vulnerability Scanners – w3af, Acunetix
3. Identity Entry Points and Attack Surface

Other domains like Cloud computing, IoT, Blockchain and edge computing were also mentioned. About how security can be provided to these.

Coding Challenges Details: (Attach the snapshot and briefly write the report for the same)

**PROGRAM 1: generating Armstrong numbers using Python programming language.**

num = int(input("Enter a number: "))

sum = 0

temp = num

**while** temp > 0:

   digit = temp % 10

   sum += digit \*\* 3

   temp //= 10

**if** num == sum:

**print**(num,"is an Armstrong number")

**else**:

**print**(num,"is not an Armstrong number")

**PROGRAM 2: Write a C Program to Reverse a Linked List in groups of given size.**

**Test Case 1:**

**If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8**

**The value of size k is 2**

**Then the linked list looks like: 2 → 1 → 4 → 3 → 6 → 5 → 8 → 7**

**Test Case 2:**

**If a linked listis: 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8**

**The value of size k is 3**

**Then the linked list looks like: 3 → 2 → 1 → 6 → 5 → 4 → 8 → 7**

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);