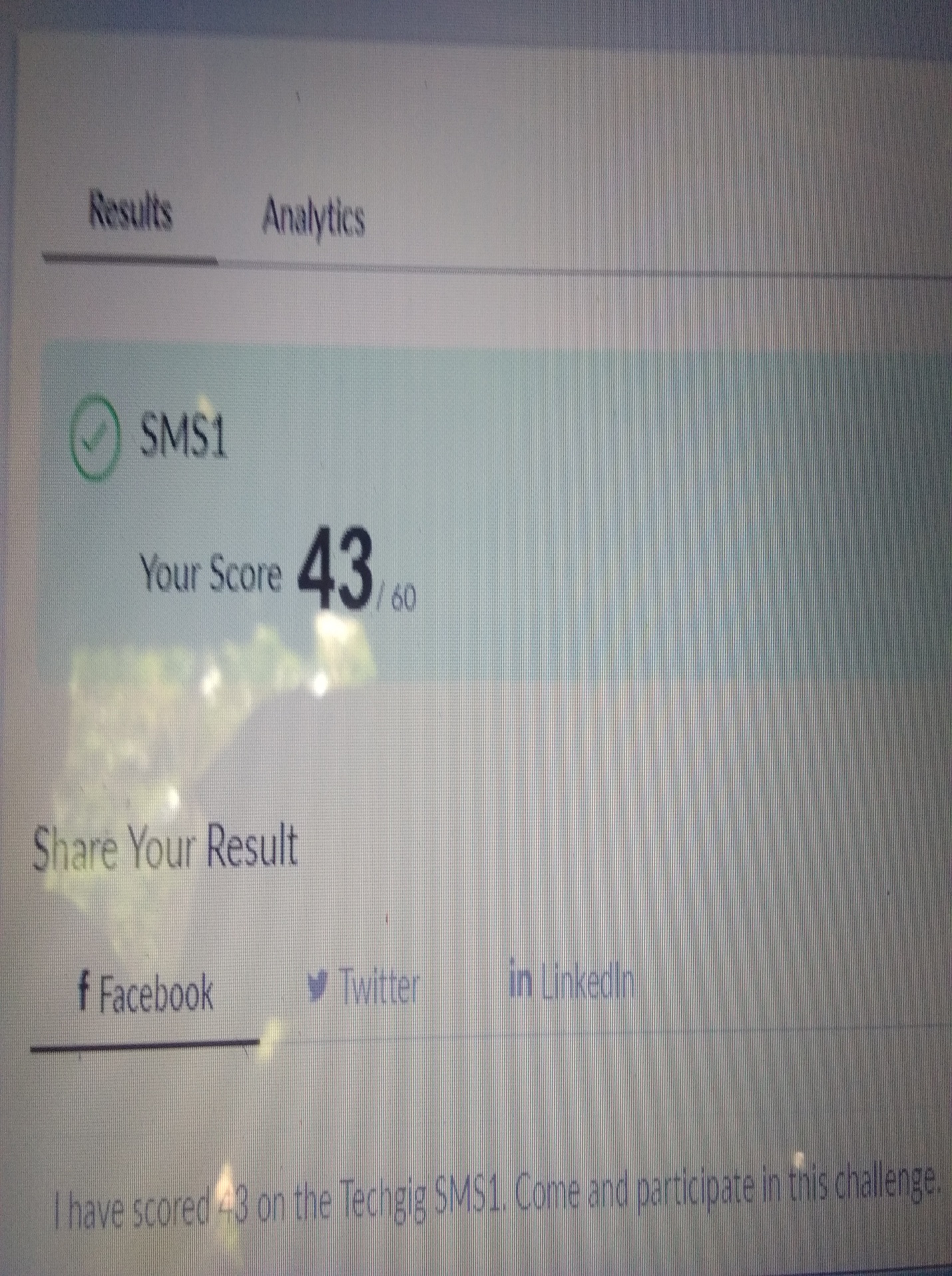
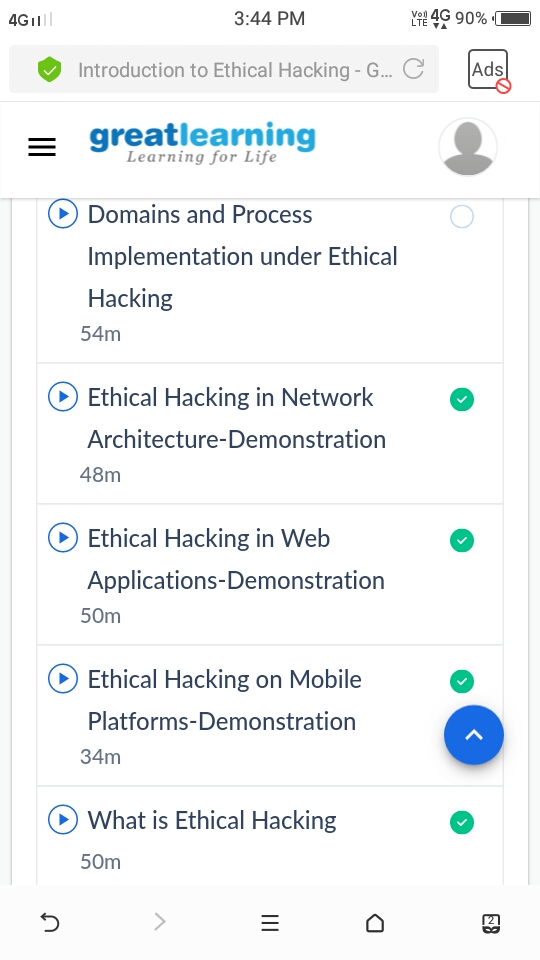
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **21-05-20** | | | | | **Name:** | **Vandana E V** | |
| **Sem & Sec** | **8th 'A'** | | | | | **USN:** | **4AL15CS103** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **System Modelling and Simulation** | | | | | | |
| **Max. Marks** | | **60** | | **Score** | | | **43** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Introduction to ethical hacking** | | | | | | | |
| **Certificate Provider** | | | **Great lerning** | | **Duration** | | | **6hrs** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:Create a SSl and then reverse the link in SSL until head becomes NULL each time reversing the link head must be moved to next immediate node** | | | | | | | | |
| **Status:Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Yes** | | | |
| **If yes Repository name** | | | | | **Vandana** | | | |
| **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) 

coding Challenges Details: (Attach the snapshot and briefly write the same

Create the SLL, and then Reverse the Link in SLL until Head becomes NULL. Each Time Reversing the Link, Head must be moved to next immediate node.

#include <stdlib.h>

struct node

{

int data;

struct node next;

};

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));

}

}

int main()

{

Struct node \*prev,\*head,\*p;

int n,i;

printf ("number of elements:");

scanf("%d",&n);

head=NULL;

for(i=0;i<n;i++)

{

p=malloc(sizeof(struct node));

scanf("%d",&p->data);

p->next=NULL;

if(head==NULL)

head=p;

else

prev->next=p;

prev=p;

}

return 0;

}