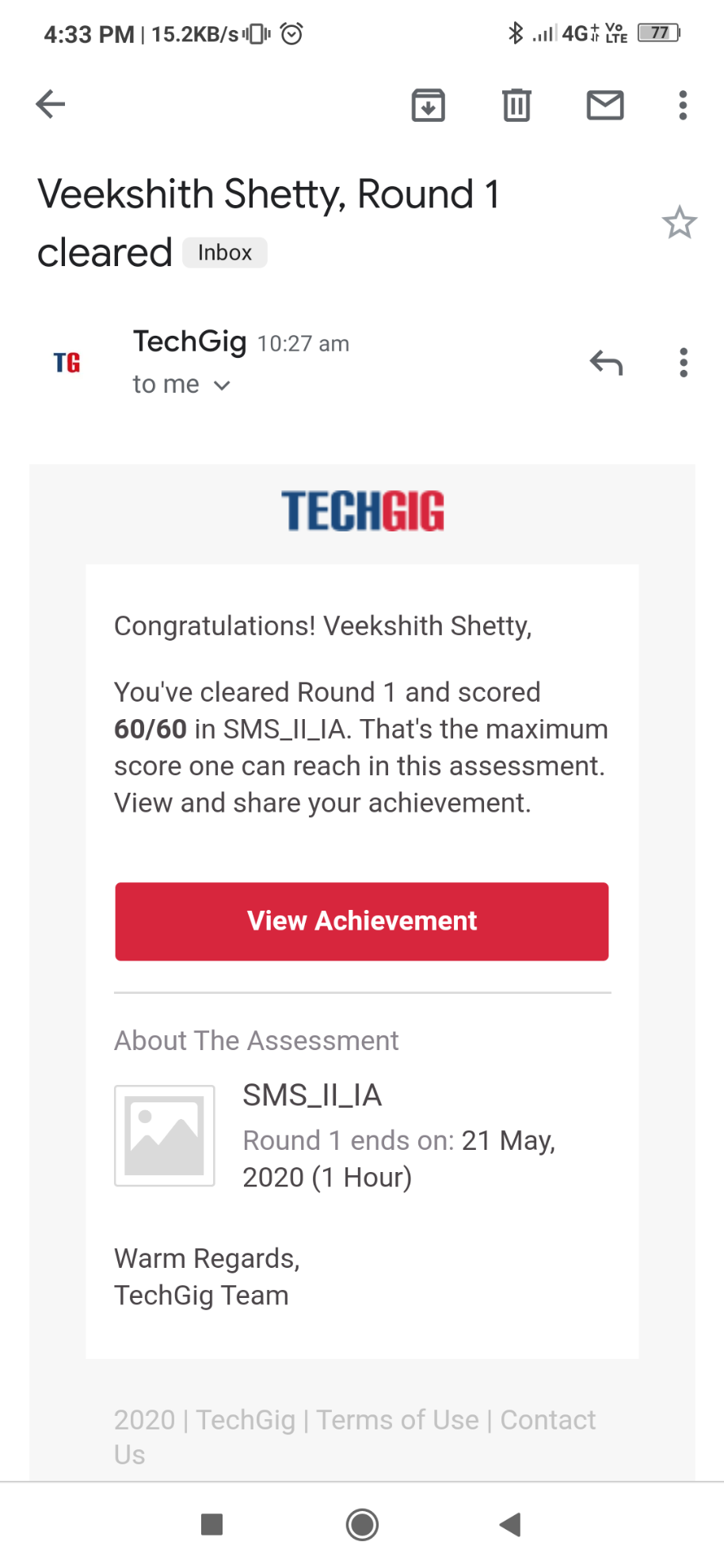
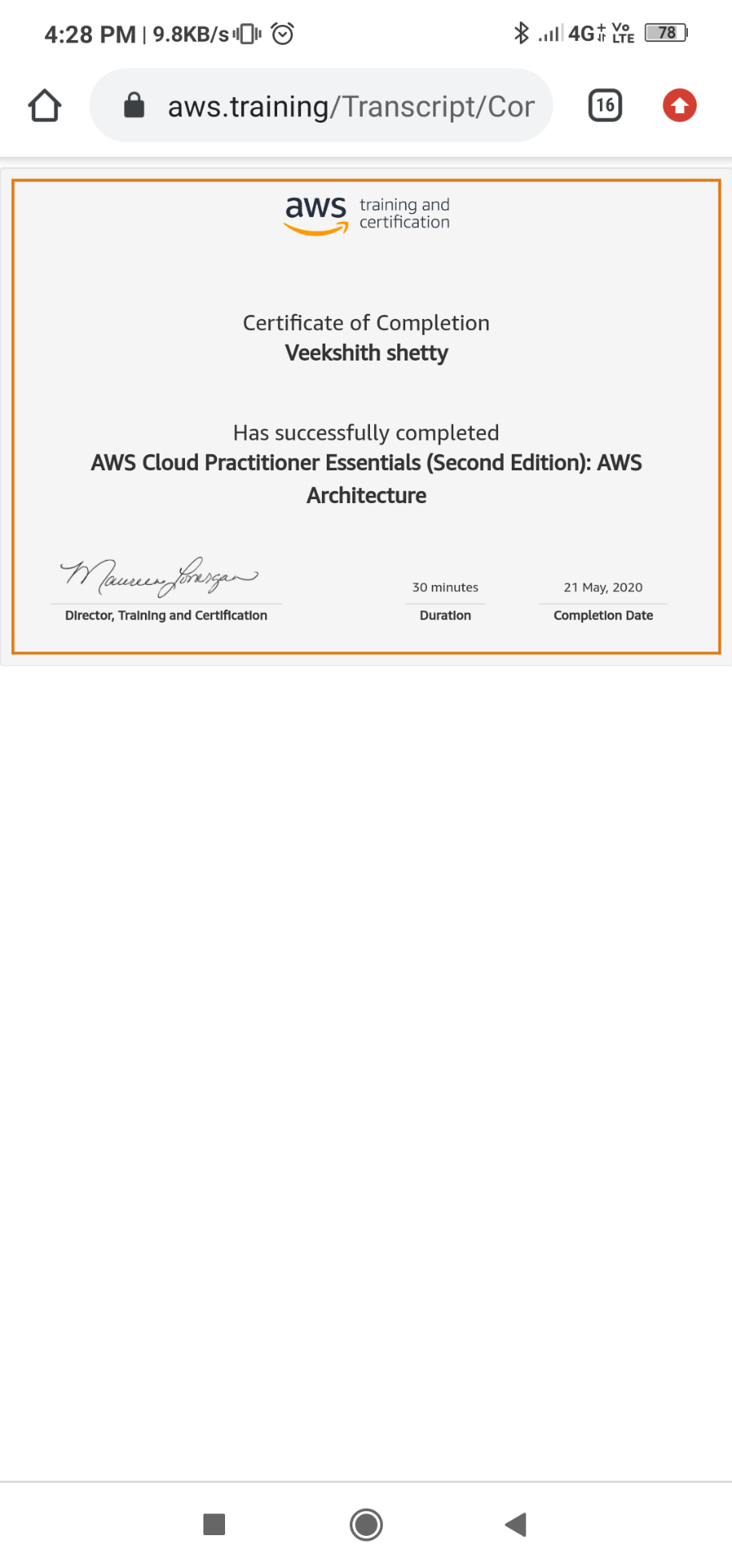
**DAILY ONLINE ACTIVITIES SUMMARY**

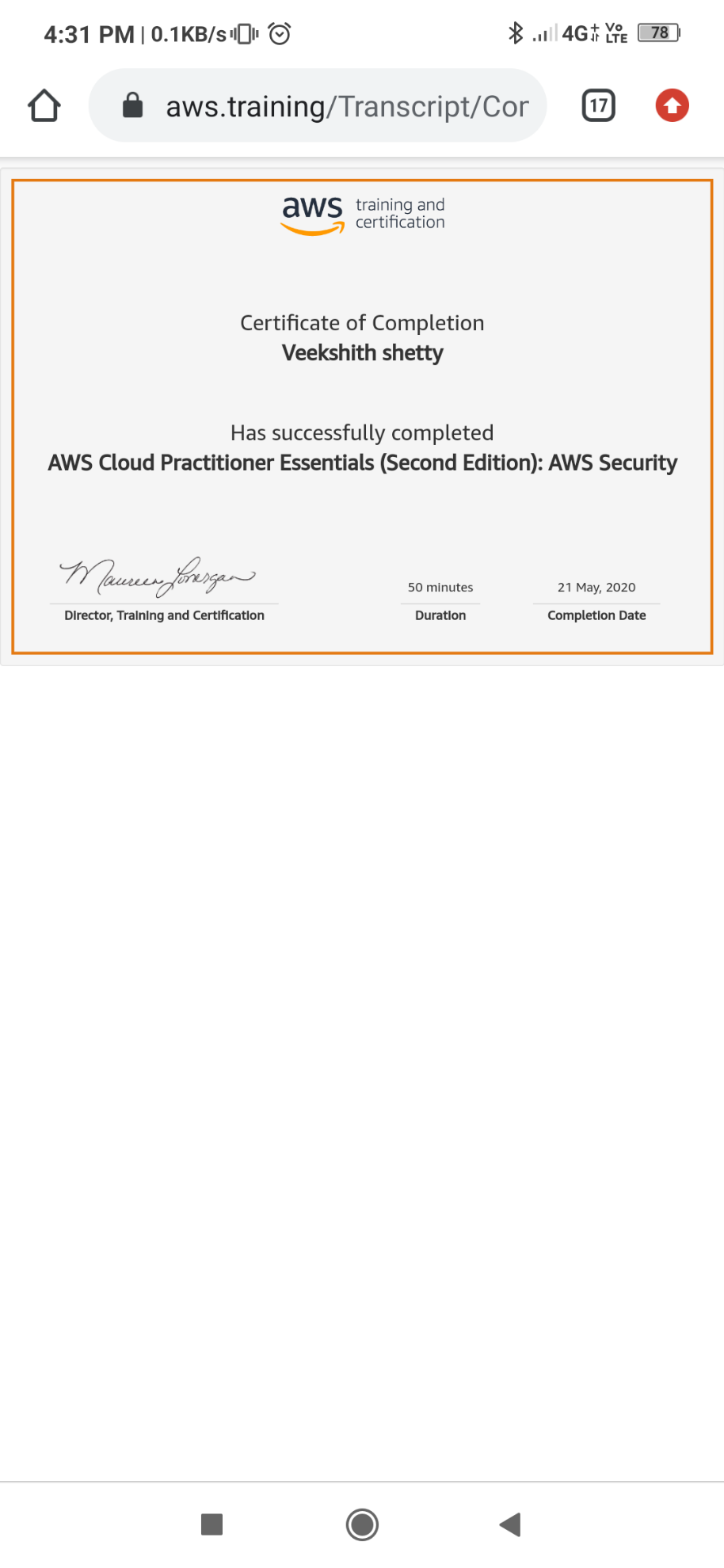
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **21/05/2020** | | | | | **Name:** | **Veekshith shetty** | |
| **Sem & Sec** | **8th B** | | | | | **USN:** | **4AL16CS097** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **SMS** | | | | | | |
| **Max. Marks** | | **60** | | **Score** | | | **60** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **AWS** | | | | | | | |
| **Certificate Provider** | | | **Amazon** | | **Duration** | | | **80 mins** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**  **prob1:Write a C Program to create a singly linked List with n elements and reverse the elements using c** | | | | | | | | |
| **Status: Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Yes** | | | |
| **If yes Repository name** | | | | | [Veekshithshetty](https://github.com/Veekshithshetty)/[codingpart](https://github.com/Veekshithshetty/codingpart) | | | |
| **Uploaded the report in slack** | | | | | **Yes** | | | |

Online Test Details:



Certification Course Details:





Coding Challenges Details:

Program 1:

#include <stdio.h>

#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; }