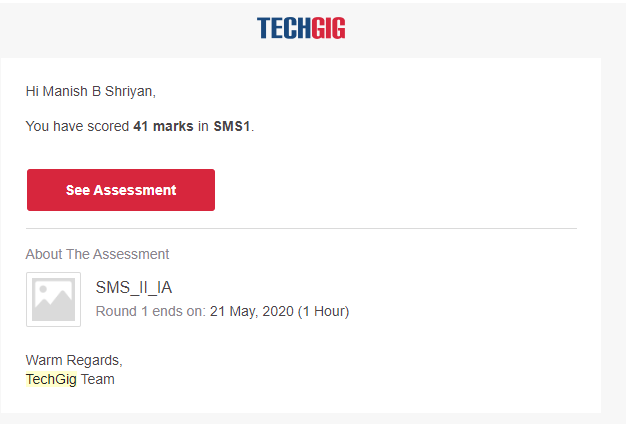
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **21/05/2020** | | | | | **Name:** | **Manish B Shriyan** | |
| **Sem & Sec** | **8th sem B sec** | | | | | **USN:** | **4AL16CS131** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **SMS** | | | | | | |
| **Max. Marks** | | **60** | | **Score** | | | **41** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **AWS Cloud Practitioner Essentials(Second Edition): AWS Core Services** | | | | | | | |
| **Certificate Provider** | | | **AWS** | | **Duration** | | | **45 Mins** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**  **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.** | | | | | | | | |
| **Status: Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Uploaded** | | | |
| **If yes Repository name** | | | | | **ManishShriyan** | | | |
| **Uploaded the report in slack** | | | | | **Yes** | | | |

Online Test Details:



Certification Course Details:



Coding Challenges Details:

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 <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;  
}