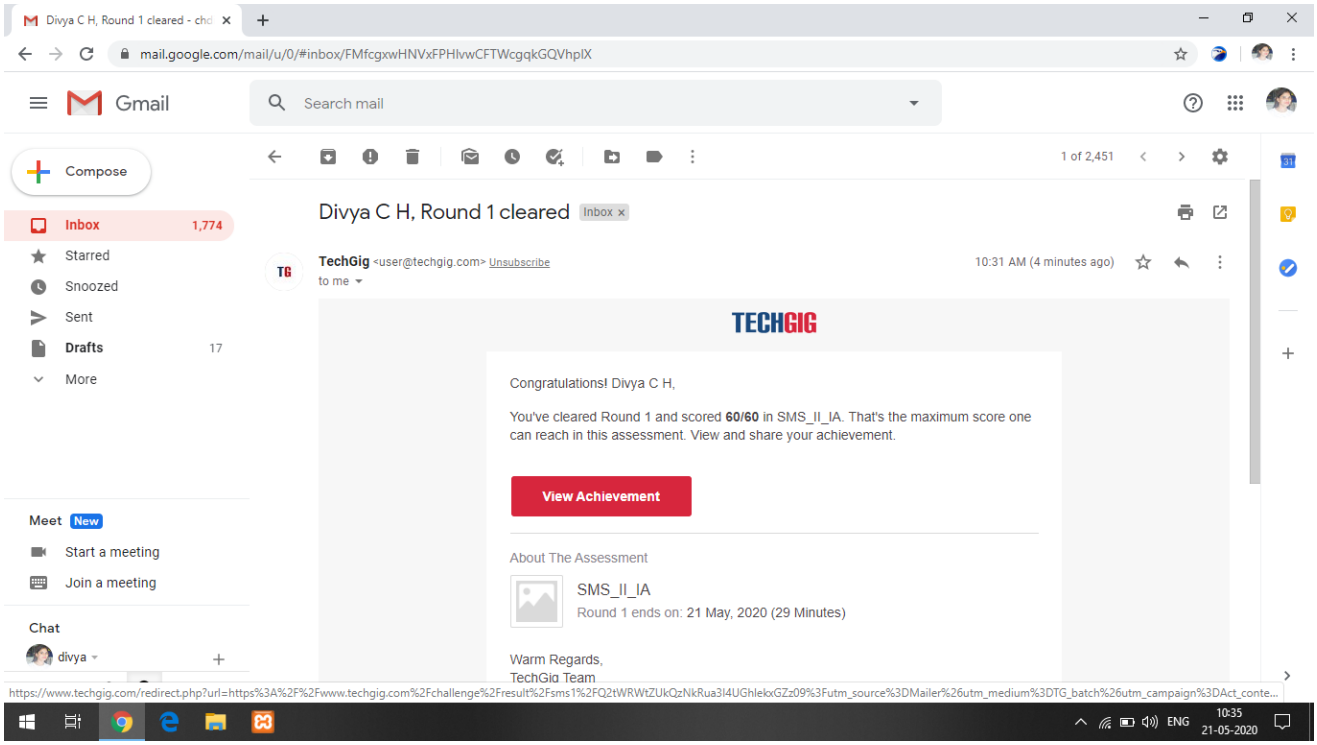
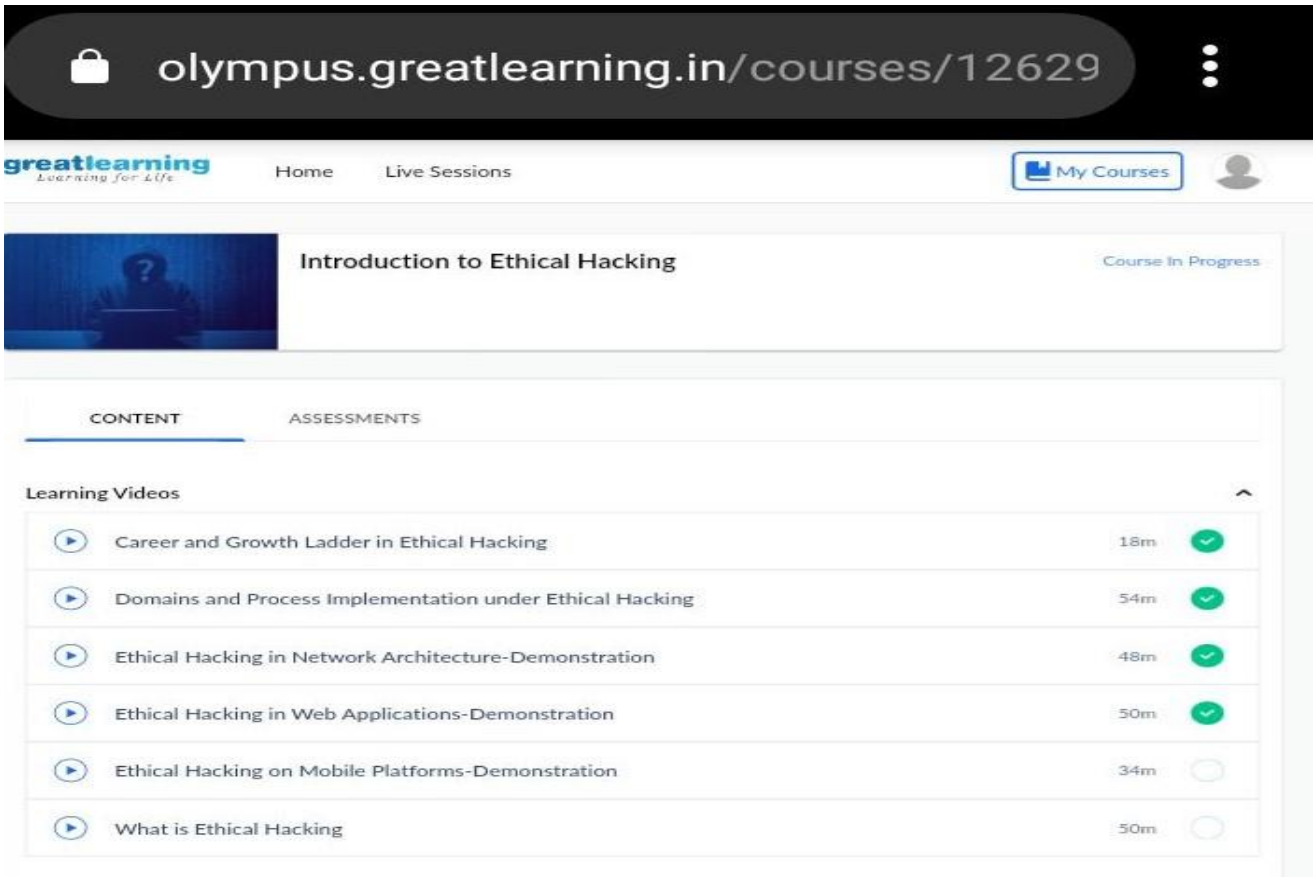
DAILY ONLINE ACTIVITIES SUMMARY

Date:	**21/5/2020**	**Name:**	**Divya C H**
Sem & Sec	**8th Sem**	**USN:**	**4AL16CS033**
Online Test Summary			
Subject	**System Model-ling and Simulation**		
Max. Marks	**60**	**Score**	**60**
Certification Course Summary			
Course	**Introduction to Ethical Hacking**		
Certificate Provider	**greatlearning.in**	**Duration**	**6 hrs**
Coding Challenges			
Problem Statement: 1) Write C Program to create Singly Liked List with n elements and reverse the elements using C 2) Python program in number right angled triangle 3) Write a menu program in Python to find Area-Circle, Circumference-Circle, Area-Square, Circumference-Square using functions with menu choice			
Status: Completed			
Uploaded the report in Github	**Yes**		
If yes Repository name	**Daily_report**		
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 num;
   struct node *nextptr;
}*stnode;

void createNodeList(intn);
void reverseDispList();
void displayList();

int main()
{
                                    int n;
                                    printf("\n\n Linked List : Create a singly linked list and
print it in reverse order :\n");

printf(" ---------------------------------------------------------- \n");

                                    printf(" Input the number of nodes : ");
                                    scanf("%d", &n);
                                    createNodeList(n);
```

```
                                printf("\n Data entered in the list are : \n");
                                displayList();
                                reverseDispList();
                                printf("\n The list in reverse are : \n");
                                displayList();
                                return 0;
}

void createNodeList(int n)
{
   struct node *fnNode, *tmp;
   int num, i;
   stnode = (struct node *)malloc(sizeof(struct node));
   if(stnode == NULL)
   {
      printf(" Memory can not be allocated.");
   }
   else
   {

      printf(" Input data for node 1 : ");
      scanf("%d", &num);
      stnode-> num = num;
      stnode-> nextptr = NULL;
      tmp = stnode;
```

```
      for(i=2; i<=n; i++)
      {
         fnNode = (struct node *)malloc(sizeof(struct node));
         if(fnNode == NULL)
         {
            printf(" Memory can not be allocated.");
            break;
         }
         else
         {
            printf(" Input data for node %d : ", i);
            scanf(" %d", &num);
            fnNode->num = num;
            fnNode->nextptr = NULL;
            tmp->nextptr = fnNode;
            tmp = tmp->nextptr;
         }
      }
   }
}

void reverseDispList()
{
   struct node *prevNode, *curNode;
```

```
    if(stnode != NULL)
    {
        prevNode = stnode;
        curNode = stnode->nextptr;
        stnode = stnode->nextptr;

        prevNode->nextptr = NULL;

        while(stnode != NULL)
        {
            stnode = stnode->nextptr;
            curNode->nextptr = prevNode;

            prevNode = curNode;
            curNode = stnode;
        }
        stnode = prevNode;
    }
}

void displayList()
{
    struct node *tmp;
    if(stnode == NULL)
```

```
  {
    printf(" No data found in the list.");
  }
  else
  {
    tmp = stnode;
    while(tmp != NULL)
    {
      printf(" Data = %d\n", tmp->num);
      tmp = tmp->nextptr;
    }
  }
}
```

Program 2:

```
rows = int(input("enter number of rows "))
        for i in range(0, rows + 1):
                for j in range(rows - i, 0, -1):
                        print(j, end=' ')
                print()
```

Program 3:

```
def AreaCircle(r):
        return 3.142*r*r


def CircumferenceCircle(r):
        return 2*3.142*r
```

```
def AreaSquare(b,h):
        return b*h

def CircumferenceSquare(h):
        return 4*h



def circle():
        r=float(input("Enter Radius Of Circle : "))
        a=AreaCircle(r)
        print("Area Of Circle: ",a)
        c=CircumferenceCircle(r)
        print("Circumference Of Circle is: ",c)
        print("\n ---------------------------------\n")
        return

def square():
        b=float(input('Enter Base Of Square : '))
        h=float(input('Enter Height Of Square : '))
        A=AreaSquare(b,h)
        print("Area Of Square is: ",A)
        CS=CircumferenceSquare(h)
        print("Circumference Of Square is: ",CS)
        print("\n ---------------------------------\n")
```

```
        return


while(1):
        n=int(input("1: CILRCLE\n2: SQUARE\n3: EXIT\n"))
        print("\n ----------------------------------\n")
        if n==1:
                circle()
        elif n==2:
                square()
        else:
                exit(0)
```