DAILY ONLINE ACTIVITIES SUMMARY

Date:	**10 June 2020**	**Name:**	**Asha Rudrappa Totagi**
Sem& Sec	**6thsem& A sec**	**USN:**	**4AL17CS015**

Online Test Summary

Subject	**Computer Graphics And Visualization**		
Max. Marks	**30**	**Score**	**27**

Certification Course Summary

Course	**Build A Game In Java**		
Certificate Provider	**Eduonix**	**Duration**	**3 hours**

Coding Challenges

Problem Statement

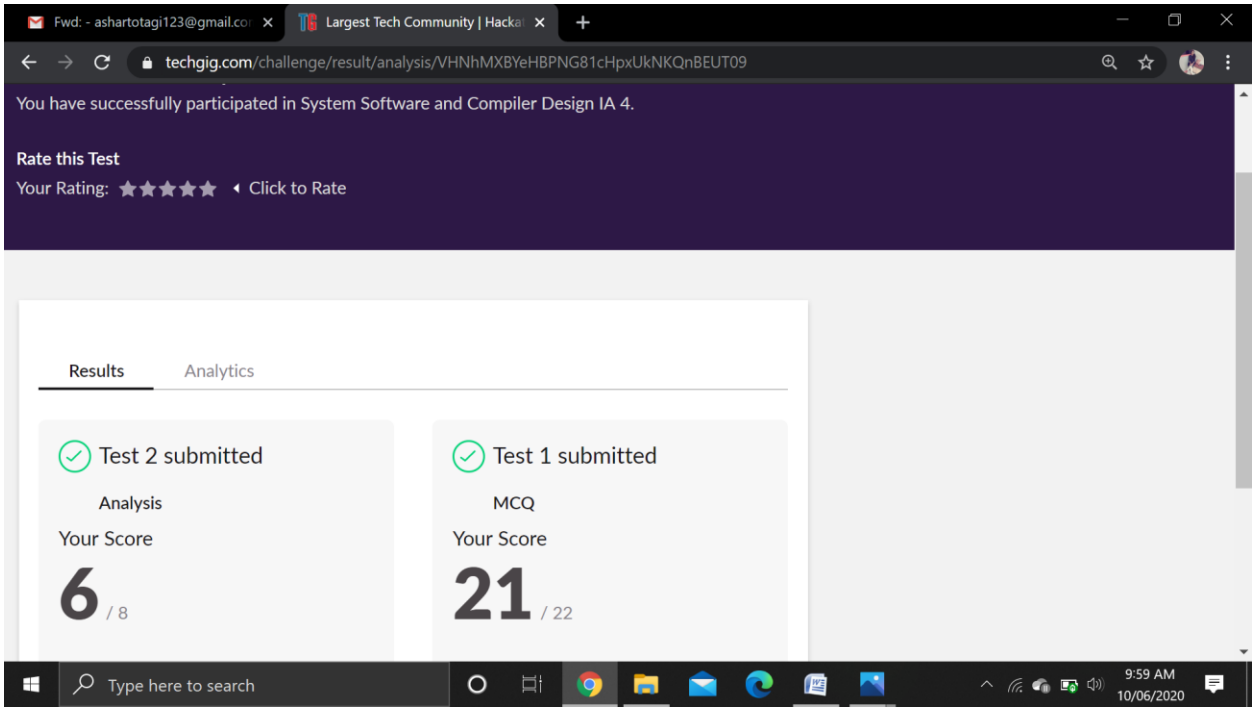
Program 1: Write a C Program to print the sum of boundary elements of a matrix
Given a matrix, the task is to print the boundary elements of the matrix and display their sum.

Program 2: Write a Java program to find the maximum and minimum value node from a circular linked list.

Status: DONE

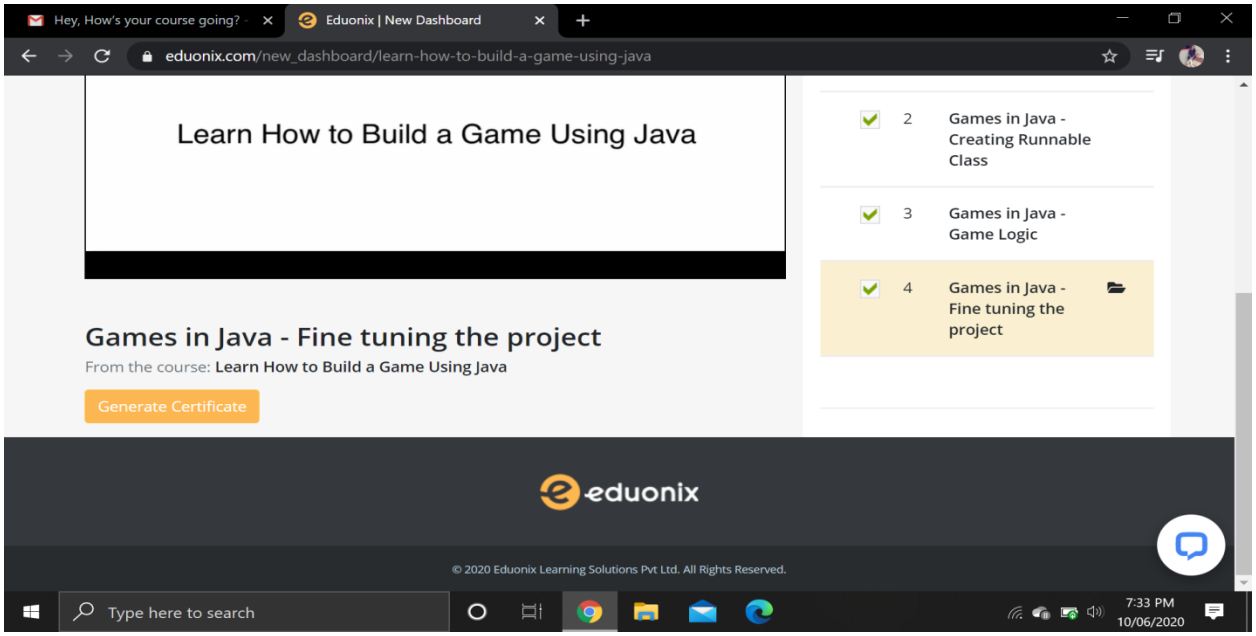
Uploaded the report in Github	**YES**
If yes Repository name	**Daily Status**
Uploaded the report in slack	**YES**

Online Test Details: (Attach the snapshot and briefly write the report for the same)



SSCD IA4 was held today i.e, 10 June 2020. Out of 30 marks I scored 27.

Certification Course Details: (Attach the snapshot and briefly write the report for the same



DAY 2 (10-06-2020) – Introduction to create a game logic.

Certificate From Eduonix

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

Program 1:

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
   int **a,r,c,i,j;
   printf("enter the size:");
   scanf("%d",&r);
   scanf("%d",&c);
   a=(int**)malloc(r*sizeof(int*));
   for(i=0;i<r;i++)
      *(a+i)=(int*)malloc(c*sizeof(int));
      printf("enter the matrix:\n");
   for(i=0;i<r;i++)
   {
      for(j=0;j<c;j++)
      {
         scanf("%d",*(a+i)+j);
      }
   }
   i=0;int sum1=0;
   for(j=0;j<c;j++)
      sum1=sum1+*(*(a+i)+j);
```

```
    i=r-1;int sum2=0;
    if(i!=0)
    {
        for(j=0;j<c;j++)
            sum2=sum2+*(*(a+i)+j);
    }
    j=0; int sum3=0;
    for(i=1;i<r-1;i++)
        sum3=sum3+*(*(a+i)+j);
    j=c-1; int sum4=0;
    for(i=1;i<r-1;i++)
        sum4=sum4+*(*(a+i)+j);
    printf("Sum of boundary is %d",sum1+sum2+sum3+sum4);
    return 0;
}
```

Program 2:

```
public class MinMax {
    public class Node{
        int data;
        Node next;
        public Node(int data) {
            this.data = data;
        }
    }
```

```
public Node head = null;
public Node tail = null;
public void add(int data){
    Node newNode = new Node(data);
    if(head == null) {
        head = newNode;
        tail = newNode;
        newNode.next = head;
    }
    else {
        tail.next = newNode;
        tail = newNode;
        tail.next = head;
    }
}
  public void minNode() {
    Node current = head;
    int min = head.data;
    if(head == null) {
        System.out.println("List is empty");
    }
    else {
         do{
            if(min > current.data) {
                min = current.data;
```

```
            }
            current= current.next;
        }while(current != head);

        System.out.println("Minimum value node in the list: "+ min);
    }
}
public void maxNode() {
    Node current = head;
    int max = head.data;
    if(head == null) {
        System.out.println("List is empty");
    }
    else {
         do{
            if(max < current.data) {
                max = current.data;
            }
            current= current.next;
           }while(current != head);
        System.out.println("Maximum value node in the list: "+ max);
    }
}
public static void main(String[] args) {
    MinMax cl = new MinMax();
```

```
        cl.add(5);
        cl.add(20);
        cl.add(10);
        cl.add(1);
        cl.minNode();
        cl.maxNode();
    }
}
```