# DAILY ONLINE ACTIVITIES SUMMARY

| | | | |
|---|---|---|---|
| **Date:** | **10-06-2020** | **Name:** | **Anvitha Poojary** |
| **Sem & Sec** | **6A** | **USN:** | **4AL17CS008** |

## Online Test Summary

| | | | |
|---|---|---|---|
| **Subject** | **SSCD** | | |
| **Max. Marks** | **30** | **Score** | **22** |

## Certification Course Summary

| | | | |
|---|---|---|---|
| **Course** | **Machine Learning with Python** | | |
| **Certificate Provider** | **COGNITIVE CLASS .ai** | **Duration** | **12hr** |

## Coding Challenges

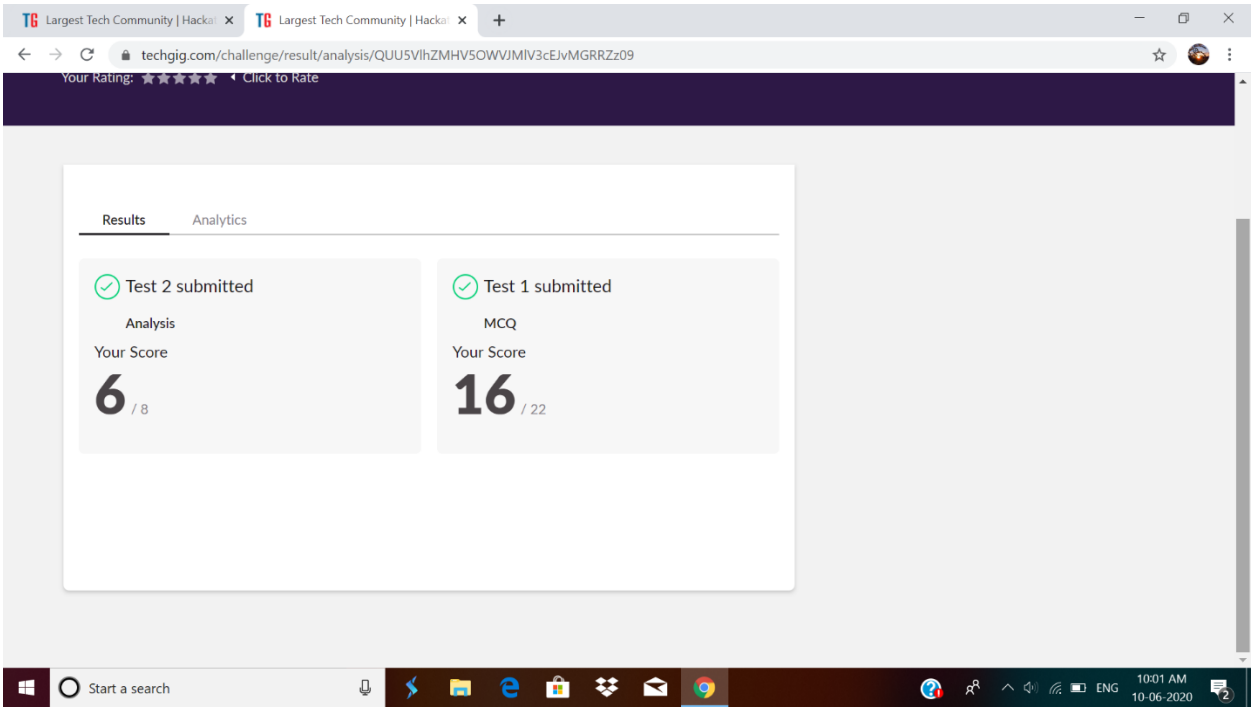
**Problem Statement:**
1.Write a C Program to print the sum of boundary elements of a matrix
2.Write a Java program to find the maximum and minimum value node from a circular linked list

**Status: completed**

| | |
|---|---|
| **Uploaded the report in Github** | **Yes** |
| **If yes Repository name** | https://github.com/anvithapo99/Daily-Report |
| **Uploaded the report in slack** | **Yes** |

## Online test details:

## Subject:SSCD

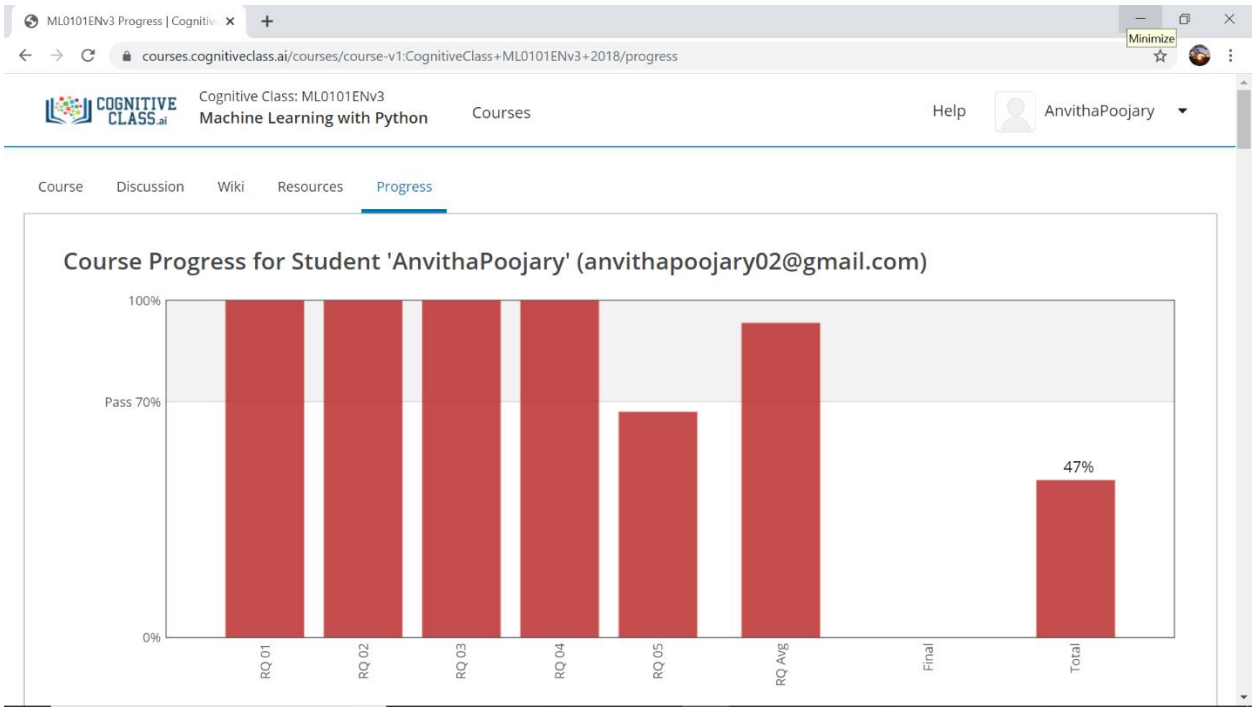



## Certification course details:

**Machine Learning with Python**

**Today I have studied following topics:**

- **Logistic regression**
- **Linear regression**
- **Training of logistic regression module**
- **What is support regression vector machine**
- **What is clustering**
- **Clustering application**




## Coding Challenges Details:

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.
Sample Output 1:
Enter M (Rows) and N (Columns): 3, 3
Enter the Elements: 1 2 3 4 5 6 7 8 9
OUTPUT:
The Input Matrix is:
1 2 3
4 5 6
7 8 9
The Boundary Elements are: 1 2 3 4 6 7 8 9
The Sum of Boundary elements of the Matrix is: 40

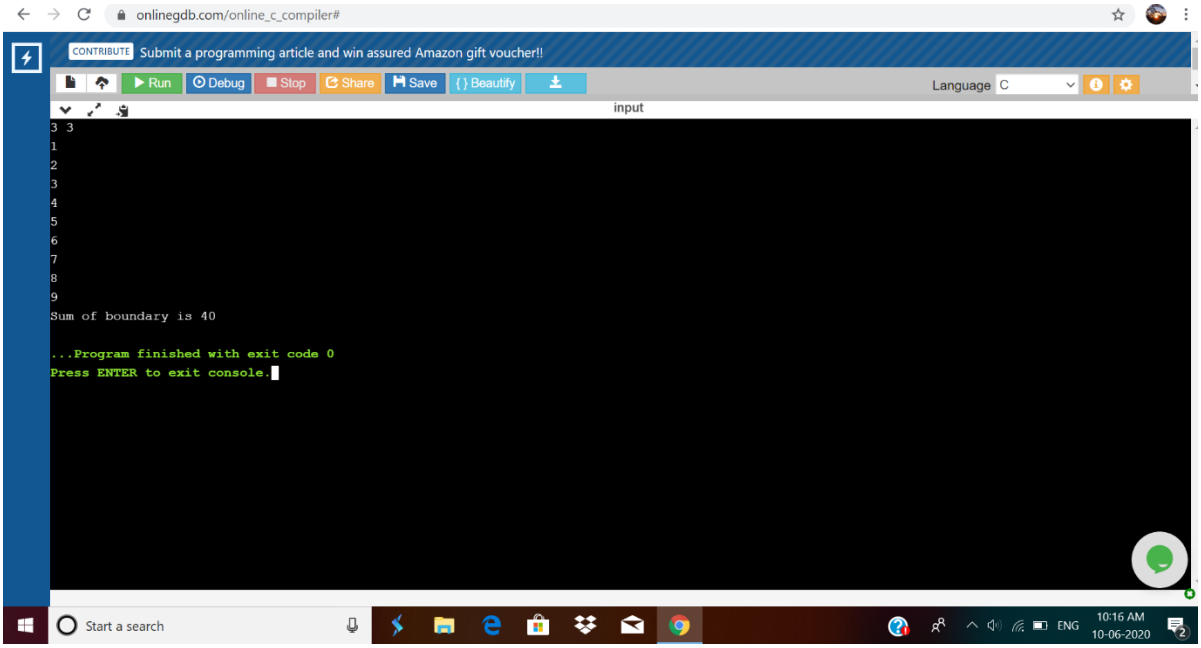
Sample Output 2:
Enter M (Rows) and N (Columns): 4, 5
Enter the Elements: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
OUTPUT:

The Input Matrix is:
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
17 18 19 20
The Boundary Elements are: 1 2 3 4 5 8 9 12 13 16 17 18 19 20
The Sum of Boundary elements of the Matrix is: 147

```
#include<stdio.h>

#include<stdlib.h>

int main()

{

   int **a,r,c,i,j;

   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));

   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;
}
```

**Output:**



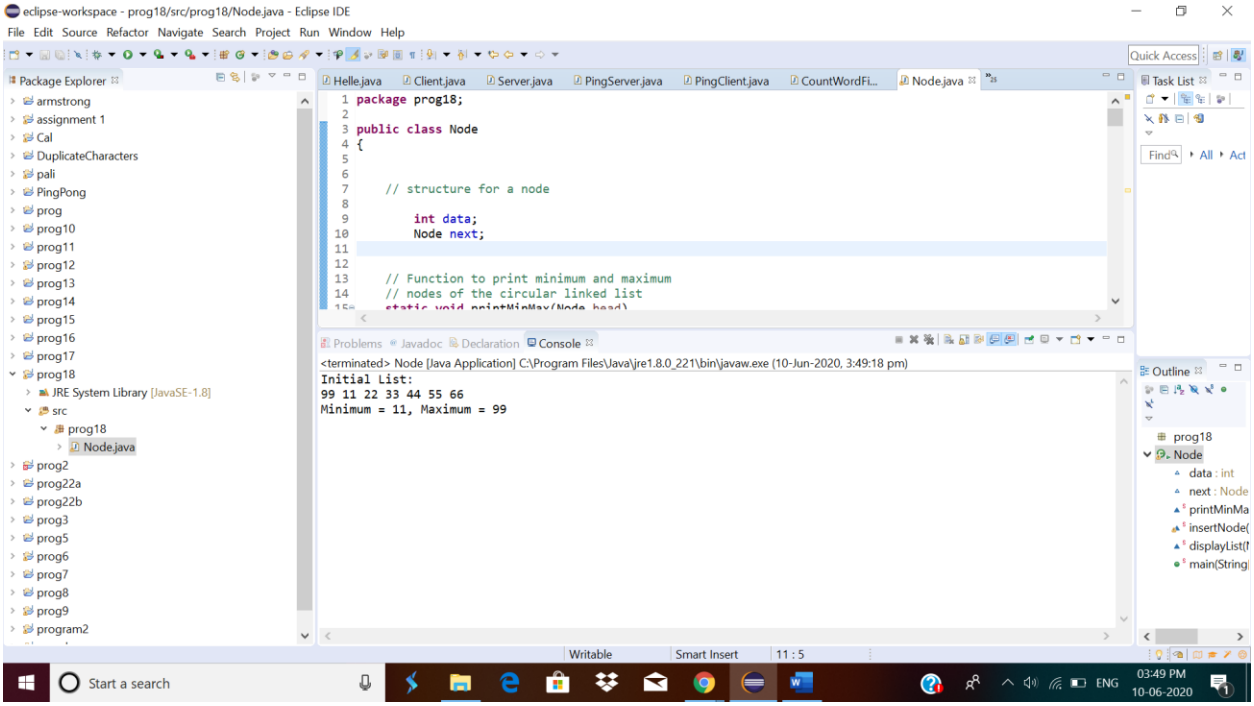

2.Write a Java program to find the maximum and minimum value node from a circular linked list

```
package prog18;

public class Node
{



        int data;
        Node next;
    static void printMinMax(Node head)
    {

        if (head == null)
        {
            return;
        }


        Node current;


        current = head;

        int min = Integer.MAX_VALUE, max = Integer.MIN_VALUE;

        while (current.next != head)
        {

            if (current.data < min)
```

```
        {
            min = current.data;
        }

        if (current.data > max)
        {
            max = current.data;
        }

        current = current.next;
    }

    System.out.println( "\nMinimum = " + min + ", Maximum = " + max);
}

static Node insertNode(Node head, int data)
{
    Node current = head;

    Node newNode = new Node();

    if (newNode == null)
    {
        System.out.printf("\nMemory Error\n");
        return null;
    }

    newNode.data = data;

    if (head == null)
    {
        newNode.next = newNode;
        head = newNode;
        return head;
    }

    else
    {

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

        newNode.next = head;

        current.next = newNode;
    }
    return head;
}

static void displayList(Node head)
{

    Node current = head;
```

```
        if (head == null)
        {
            System.out.printf("\nDisplay List is empty\n");
            return;
        }

        else
        {
            do
            {
                System.out.printf("%d ", current.data);
                current = current.next;
            } while (current != head);
        }
    }

    public static void main(String args[])
    {
        Node Head = null;

        Head=insertNode(Head, 99);
        Head=insertNode(Head, 11);
        Head=insertNode(Head, 22);
        Head=insertNode(Head, 33);
        Head=insertNode(Head, 44);
        Head=insertNode(Head, 55);
        Head=insertNode(Head, 66);

        System.out.println("Initial List: ");

        displayList(Head);

        printMinMax(Head);
    }
    }
```

Output:


eclipse-workspace - prog18/src/prog18/Node.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
Package Explorer
armstrong
assignment 1
Cal
DuplicateCharacters
pali
PingPong
prog
prog10
prog11
prog12
prog13
prog14
prog15
prog16
prog17
prog18
JRE System Library [JavaSE-1.8]
src
prog18
Node.java
prog2
prog22a
prog22b
prog3
prog5
prog6
prog7
prog8
prog9
program2
Helle.java
Client.java
Server.java
PingServer.java
PingClient.java
CountWordFi...
Node.java
1 package prog18;
2
3 public class Node
4 {
5
6
7 // structure for a node
8
9 int data;
10 Node next;
11
12
13 // Function to print minimum and maximum
14 // nodes of the circular linked list
Task List
Find
All
Act
Problems
Javadoc
Declaration
Console
<terminated> Node [Java Application] C:\Program Files\Java\jre1.8.0_221\bin\javaw.exe (10-Jun-2020, 3:49:18 pm)
Initial List:
99 11 22 33 44 55 66
Minimum = 11, Maximum = 99
Outline
prog18
Node
data : int
next : Node
printMinMa
insertNode(
displayList(
main(String
Writable
Smart Insert
11 : 5
Start a search
ENG
03:49 PM
10-06-2020