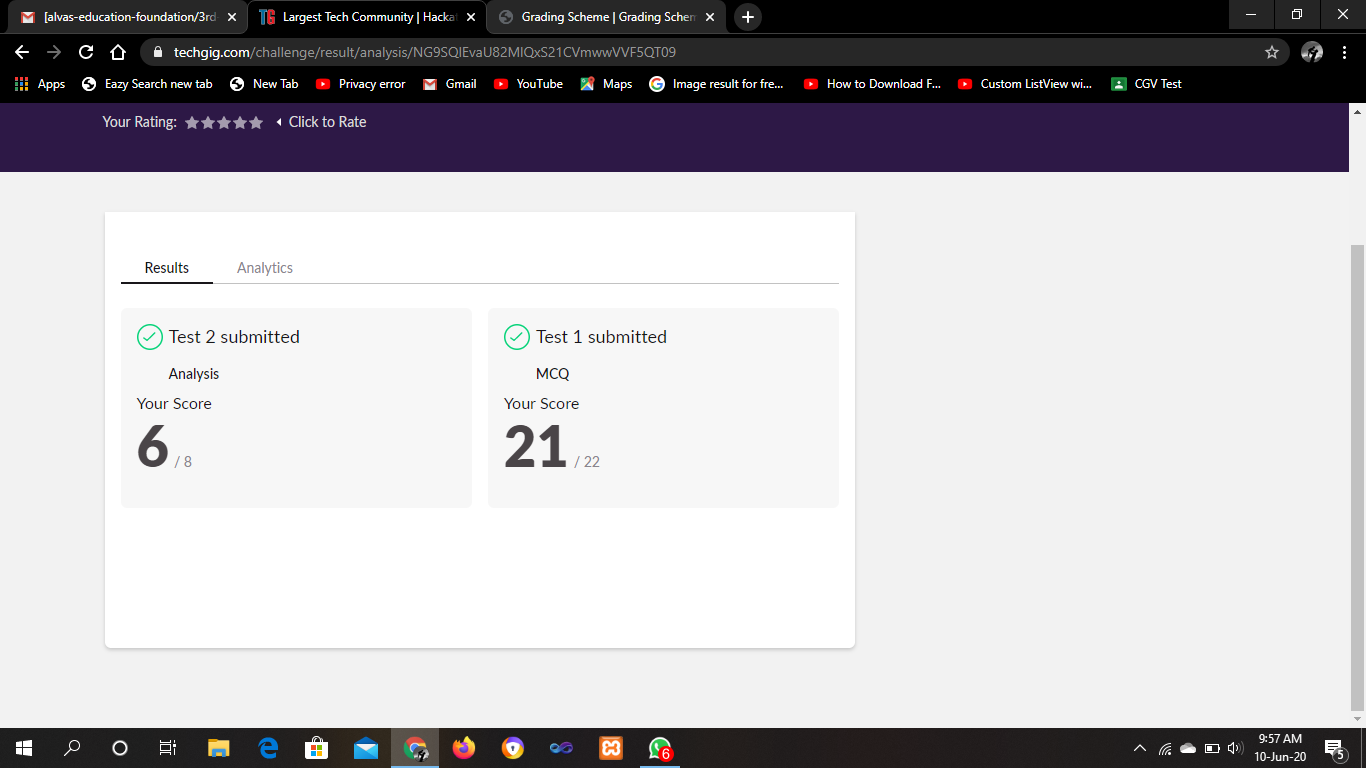
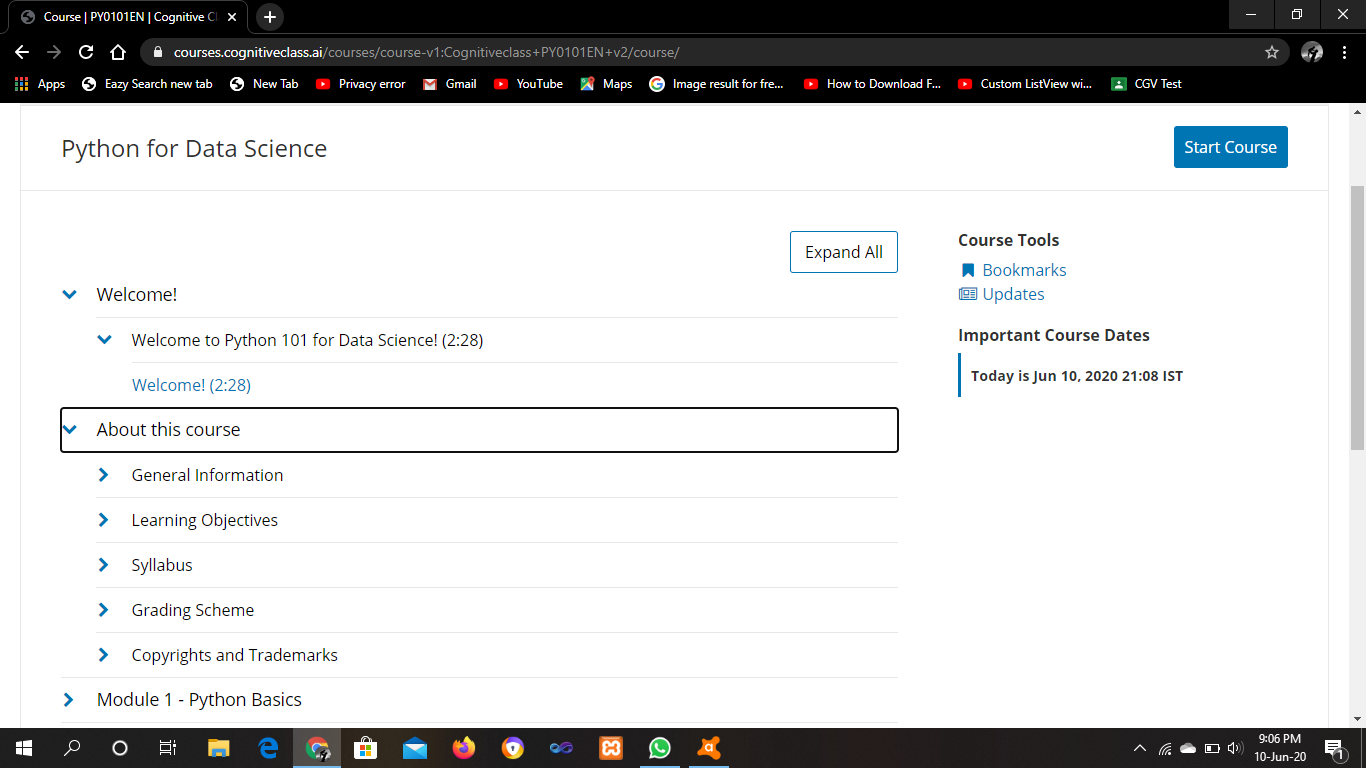
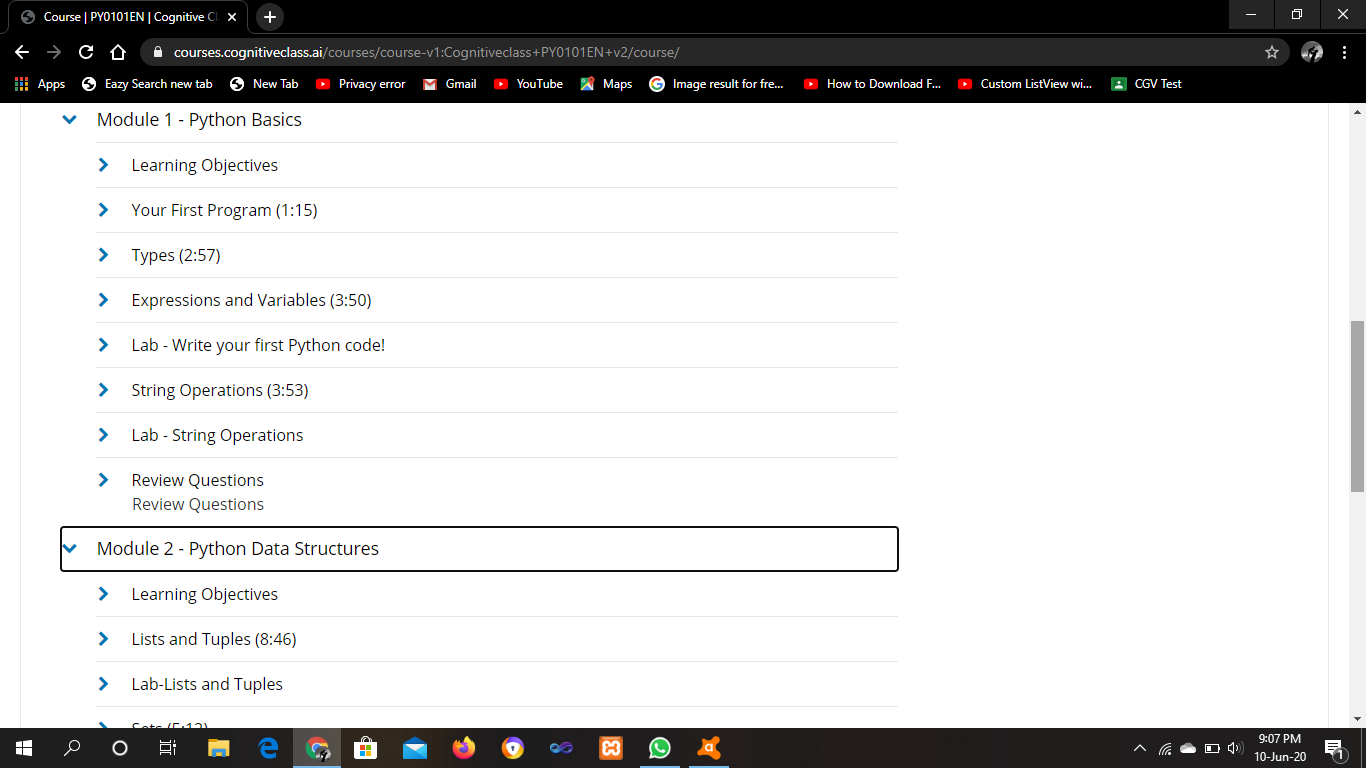
**DAILY ONLINE ACTIVITIES SUMMARY**

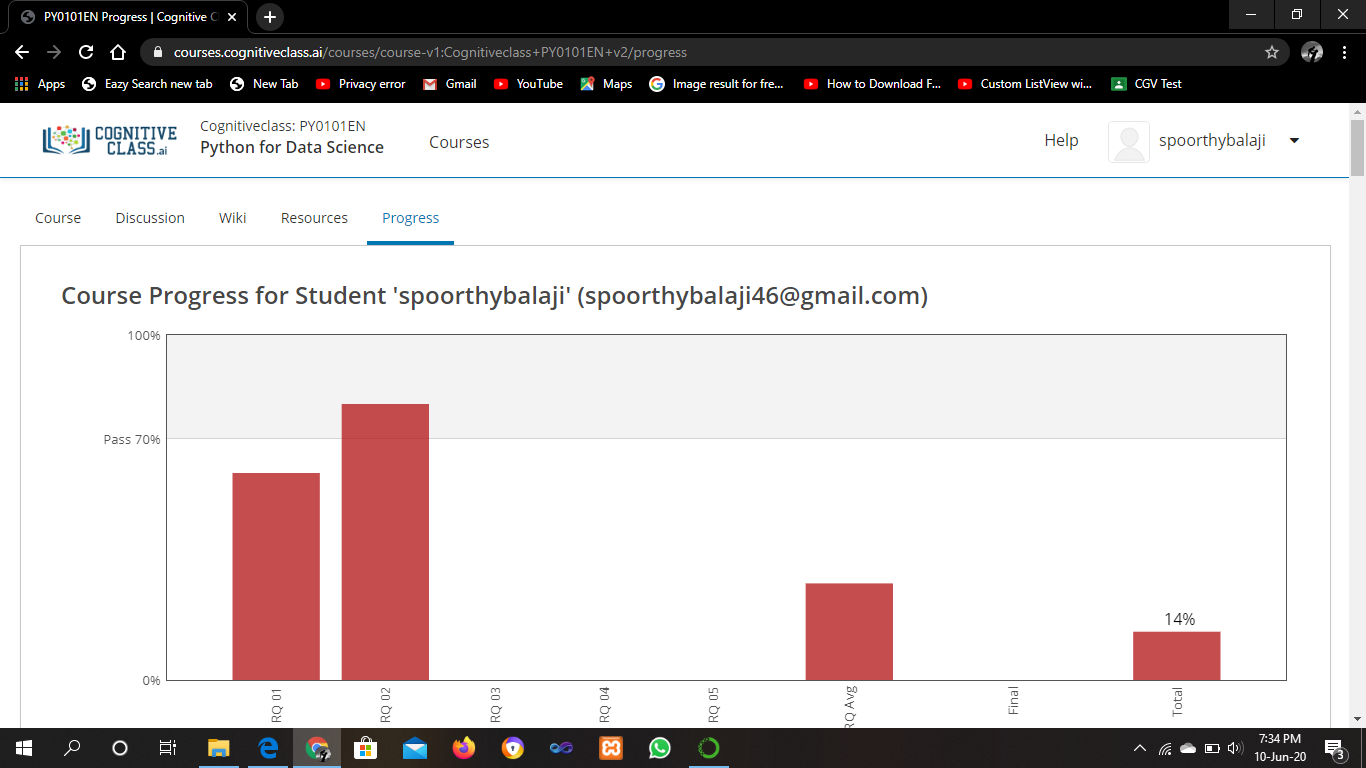
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **10/06/2020** | | | | | **Name:** | **Spoorthy Balaji** | |
| **Sem & Sec** | **6th & B** | | | | | **USN:** | **4al17cs098** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **OS** | | | | | | |
| **Max. Marks** | | **30** | | **Score** | | | **27** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Python for Data Science** | | | | | | | |
| **Certificate Provider** | | | **COGNITIVE CLASS.ia** | | **Duration** | | | **9hours** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:** 4 Programs | | | | | | | | |
| **Status: Solved** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **yes** | | | |
| **If yes Repository name** | | | | | <https://github.com/spoorthybalaji/Daily_Status> | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

IA TEST



CERTIFICATION COURSE





ONLINE CODING

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

public class MinMax {

//Represents the node of list.

public class Node{

int data;

Node next;

public Node(int data) {

this.data = data;

}

}

//Declaring head and tail pointer as null.

public Node head = null;

public Node tail = null;

//This function will add the new node at the end of the list.

public void add(int data){

//Create new node

Node newNode = new Node(data);

//Checks if the list is empty.

if(head == null) {

//If list is empty, both head and tail would point to new node.

head = newNode;

tail = newNode;

newNode.next = head;

}

else {

//tail will point to new node.

tail.next = newNode;

//New node will become new tail.

tail = newNode;

//Since, it is circular linked list tail will points to head.

tail.next = head;

}

}

//Finds out the minimum value node in the list

public void minNode() {

Node current = head;

//Initializing min to initial node data

int min = head.data;

if(head == null) {

System.out.println("List is empty");

}

else {

do{

//If current node's data is smaller than min

//Then replace value of min with current node's data

if(min > current.data) {

min = current.data;

}

current= current.next;

}while(current != head);

System.out.println("Minimum value node in the list: "+ min);

}

}

//Finds out the maximum value node in the list

public void maxNode() {

Node current = head;

//Initializing max to initial node data

int max = head.data;

if(head == null) {

System.out.println("List is empty");

}

else {

do{

//If current node's data is greater than max

//Then replace value of max with current node's data

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();

//Adds data to the list

cl.add(5);

cl.add(20);

cl.add(10);

cl.add(1);

//Prints the minimum value node in the list

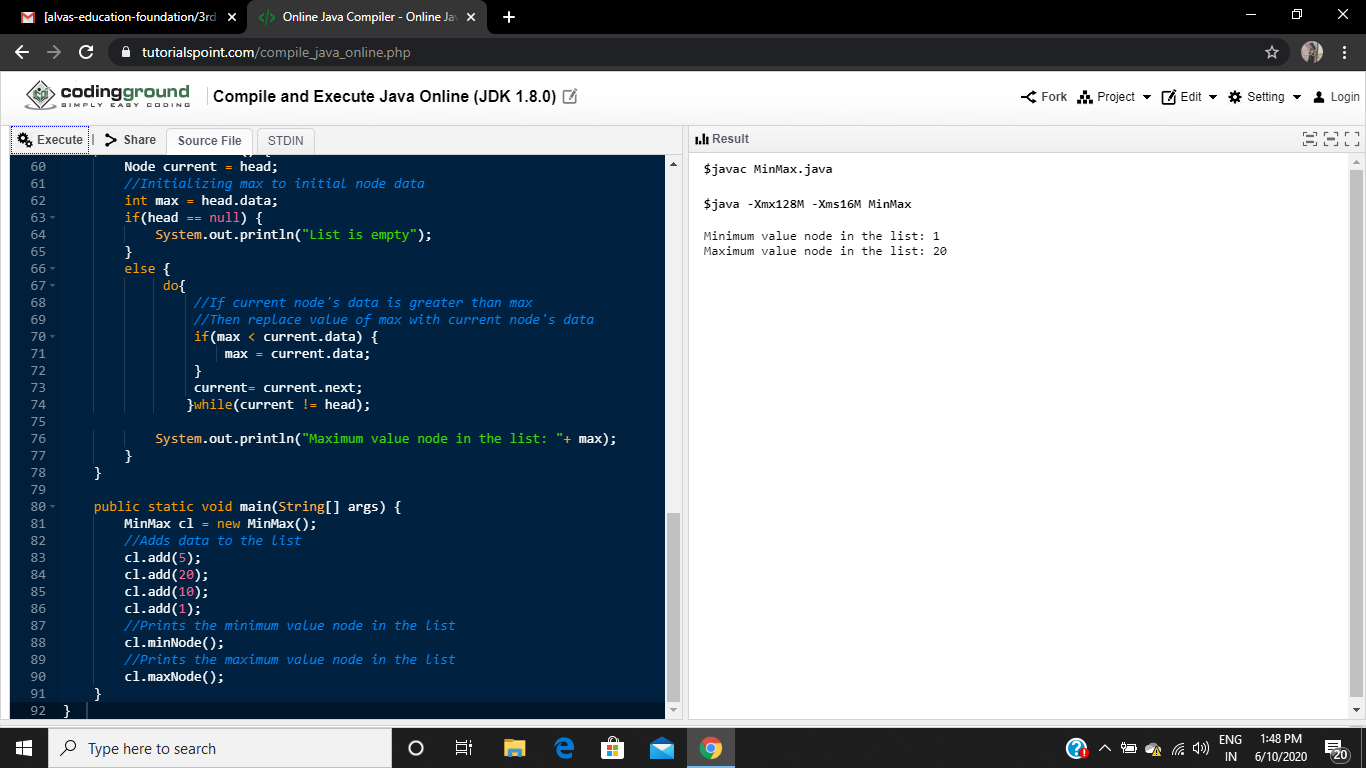
cl.minNode();

//Prints the maximum value node in the list

cl.maxNode();

}

}



2. Python program to find the length of the list using recursion

def length(lst):

if not lst:

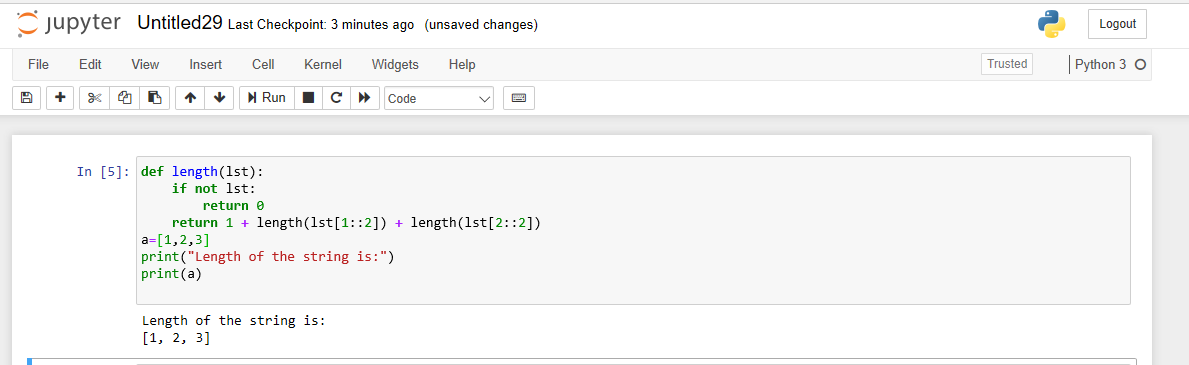
return 0

return 1 + length(lst[1::2]) + length(lst[2::2])

a=[1,2,3]

print("Length of the string is:")

print(a)



**3. Write a C Program to print the sum of boundary elements of a matrix.**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int a[100][100],m,n,i,j;

printf("Enter The Size Of The Matrix:\n");

scanf("%d%d",&m,&n);

printf("Enter The Elements Into Matrix:\n");

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

{

scanf("%d",&a[i][j]);

}

}

int f,g;

printf("The Input Matrix Is:\n");

for(f=0;f<m;f++)

{

for(g=0;g<n;g++)

{

printf("%d\t",a[f][g]);

}

printf("\n");

}

printf("The Boundary Elements Are:\n");

int b,c,s=0;

for(b=0;b<m;b++)

{

for(c=0;c<n;c++)

{

if(b==0 || b==m-1)

{

s=s+a[b][c];

printf("%d\t",a[b][c]);

}

else if(c==0 || c==n-1)

{

s=s+a[b][c];

printf("%d\t",a[b][c]);

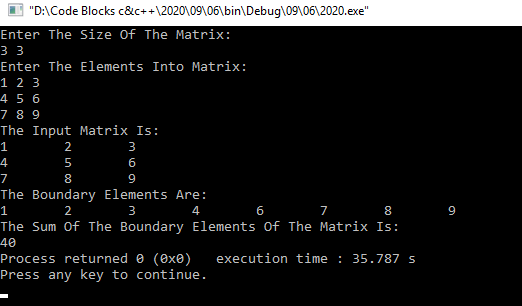
}

}

}

printf("\nThe Sum Of The Boundary Elements Of The Matrix Is:\n%d",s);

}



4. Python Program to check whether a given number is a fibonacci number or not

import math

def checkPerfectSquare(n):

    sqrt = int(math.sqrt(n))

    if pow(sqrt, 2) == n:

        return True

    else:

        return False

def isFibonacciNumber(n):

    res1 = 5 \* n \* n + 4

    res2 = 5 \* n \* n - 4

    if checkPerfectSquare(res1) or checkPerfectSquare(res2):

        return True

    else:

        return False

num = int(input("Enter an integer number: "))

if isFibonacciNumber(num):

    print ("Yes,", num, "is a Fibonacci number")

else:

    print ("No,", num, "is not a Fibonacci number")

