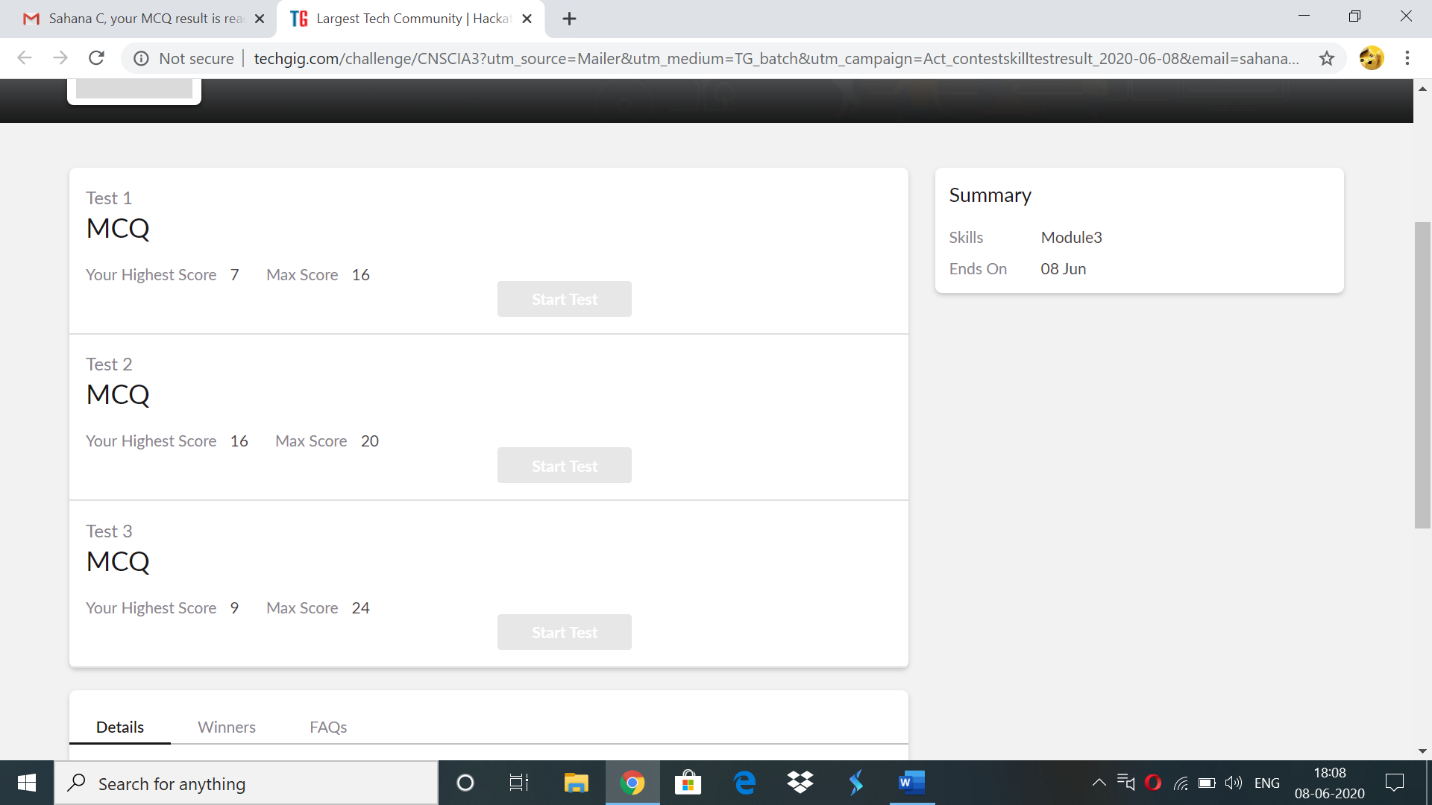
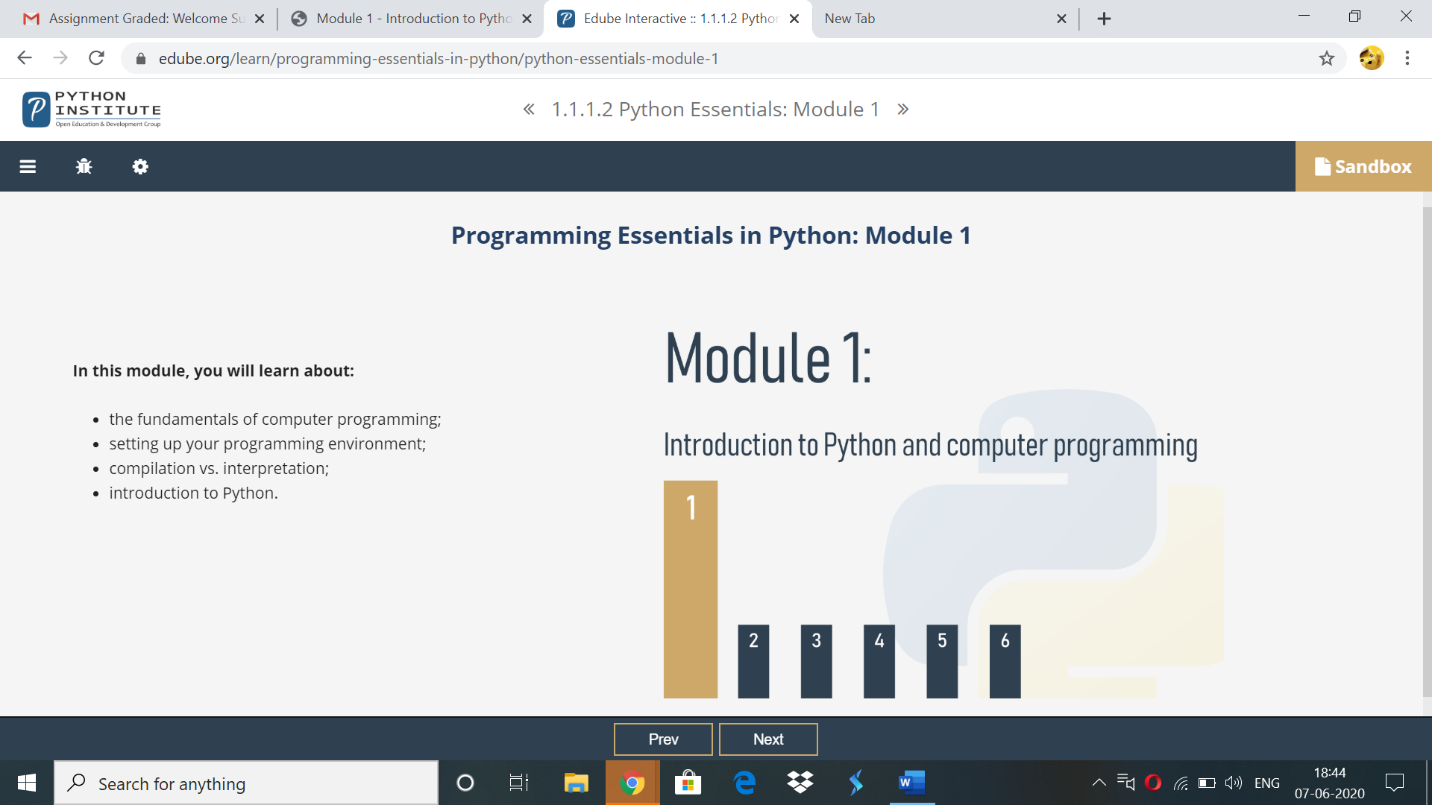
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **08-06-20** | | | | **Name:** | **SAHANA C** | |
| **Sem & Sec** | **VI B** | | | | **USN:** | **4AL17CS116** | |
| **Online Test Summary** | | | | | | | |
| **Subject** | | **CNSC IA** | | | | | |
| **Max. Marks** | | **30** | | **Score** | | **32** | |
| **Certification Course Summary** | | | | | | | |
| **Course** | Programming Essentials in Python | | | | | | |
| **Coding Challenges**  1)Write a python function that will take a string and checks whether it is a palindrome or not. Return If it a palindrome, print true else print false  2) C Program to Generate All the Set Partitions of n Numbers Beginning from 1 and so on  3) Program to Check whether a number can be represented as difference of two squares | | | | | | | |
| **Certificate Provider** | | | **Cisco -python institution** | **Duration** | | | **No limit** |
| **Status:on going** | | | | | | | |
| **Uploaded the report in Github** | | | | **Yes** | | | |
| **If yes Repository name** | | | | **https://github.com/sahanasanu/Daliy-status** | | | |
| **Uploaded the report in slack** | | | | **Yes** | | | |

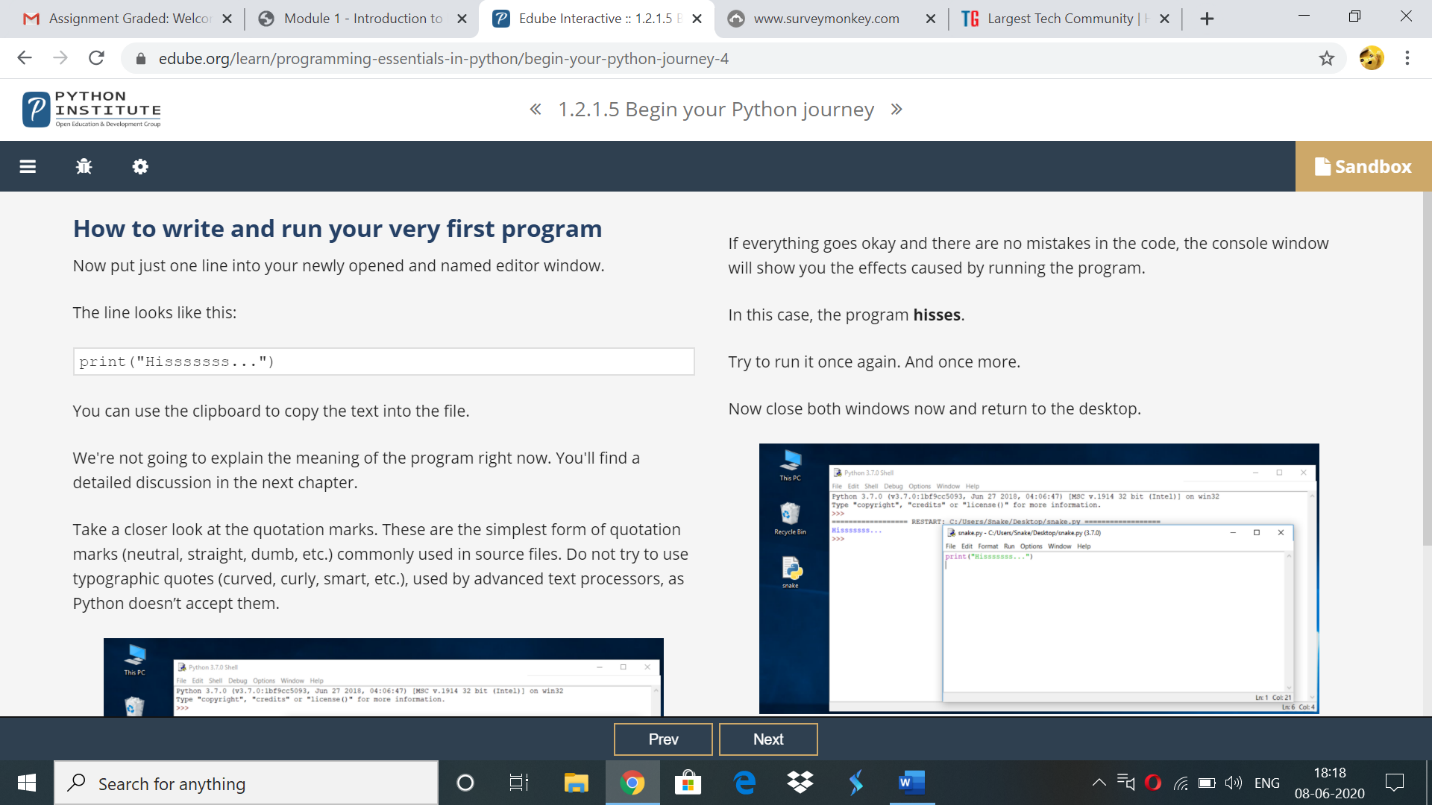
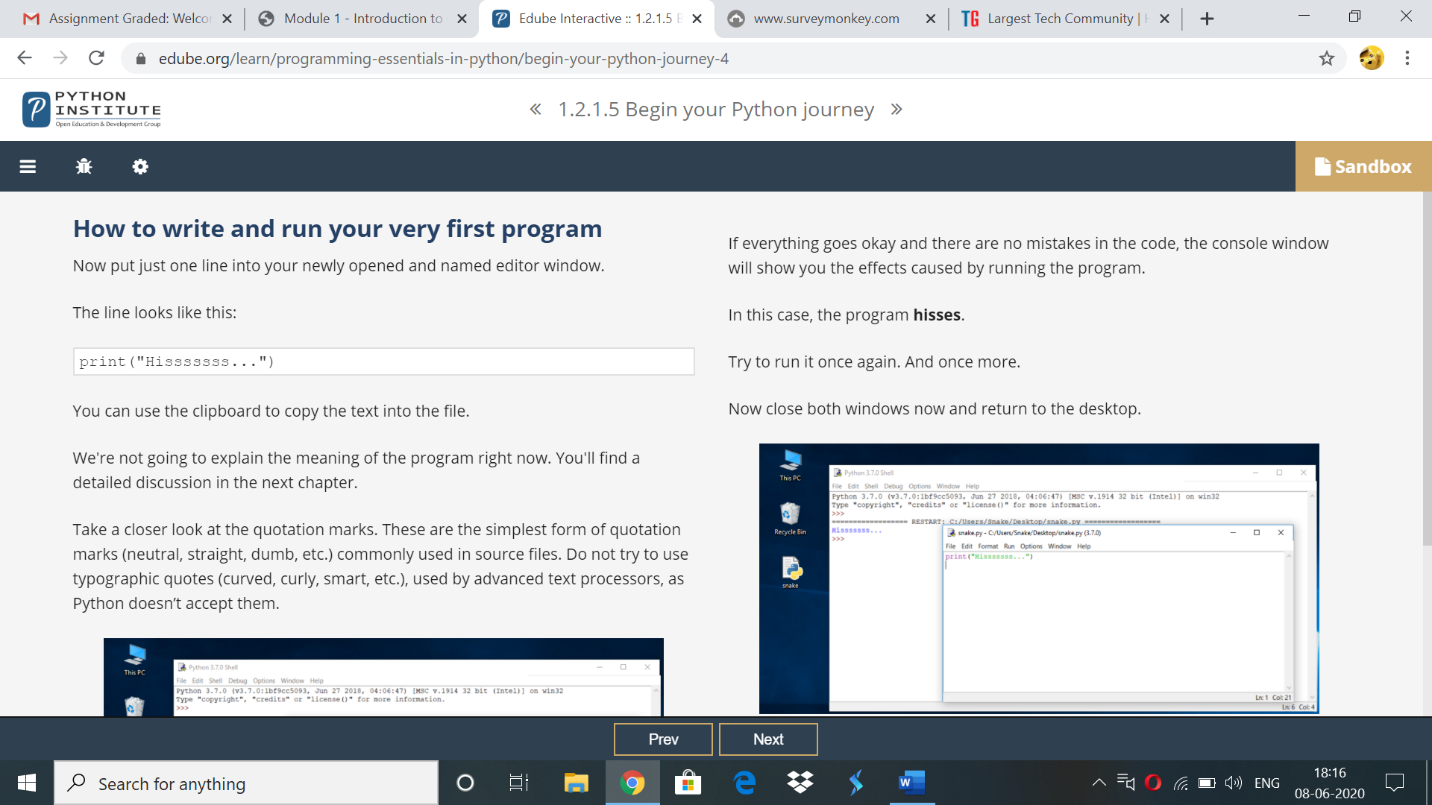
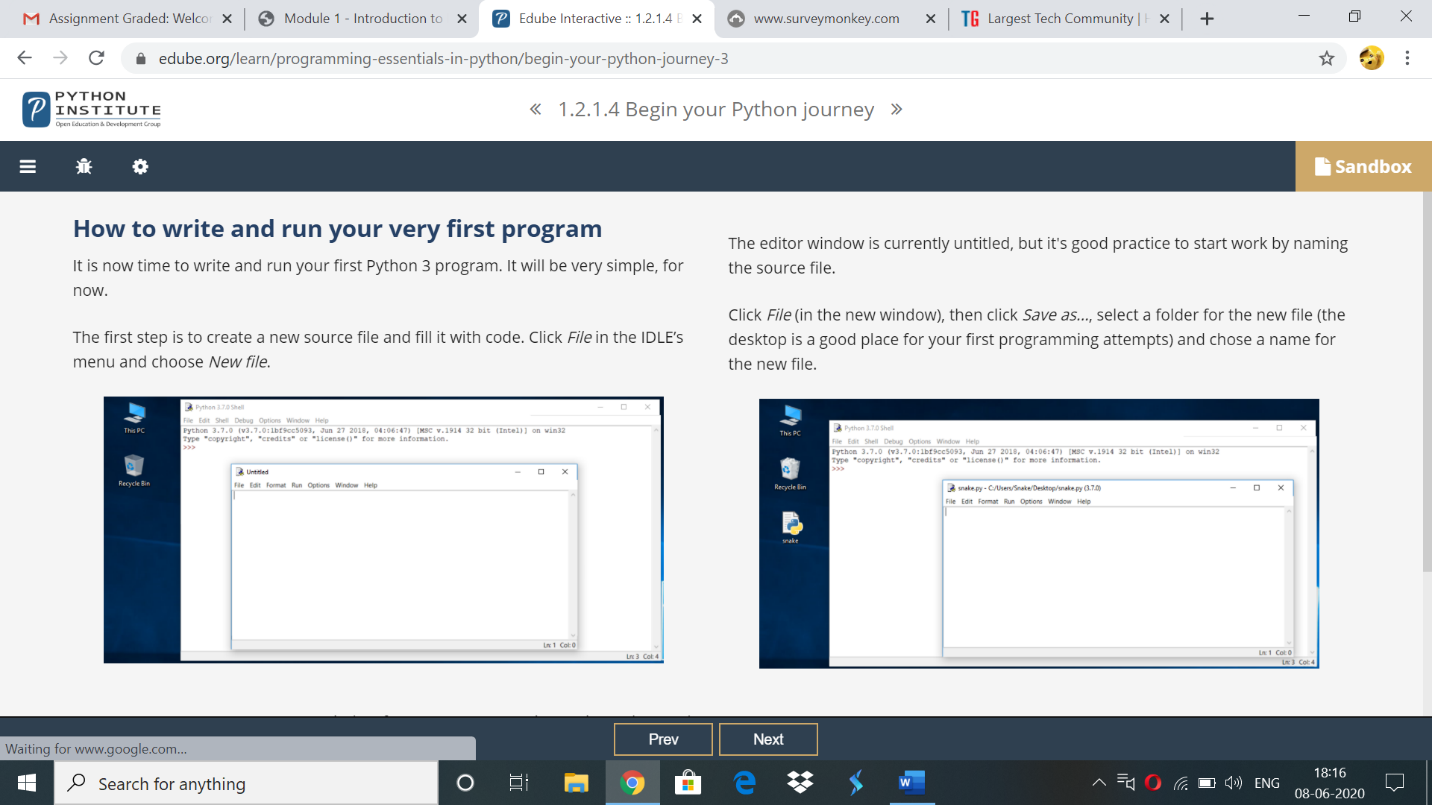
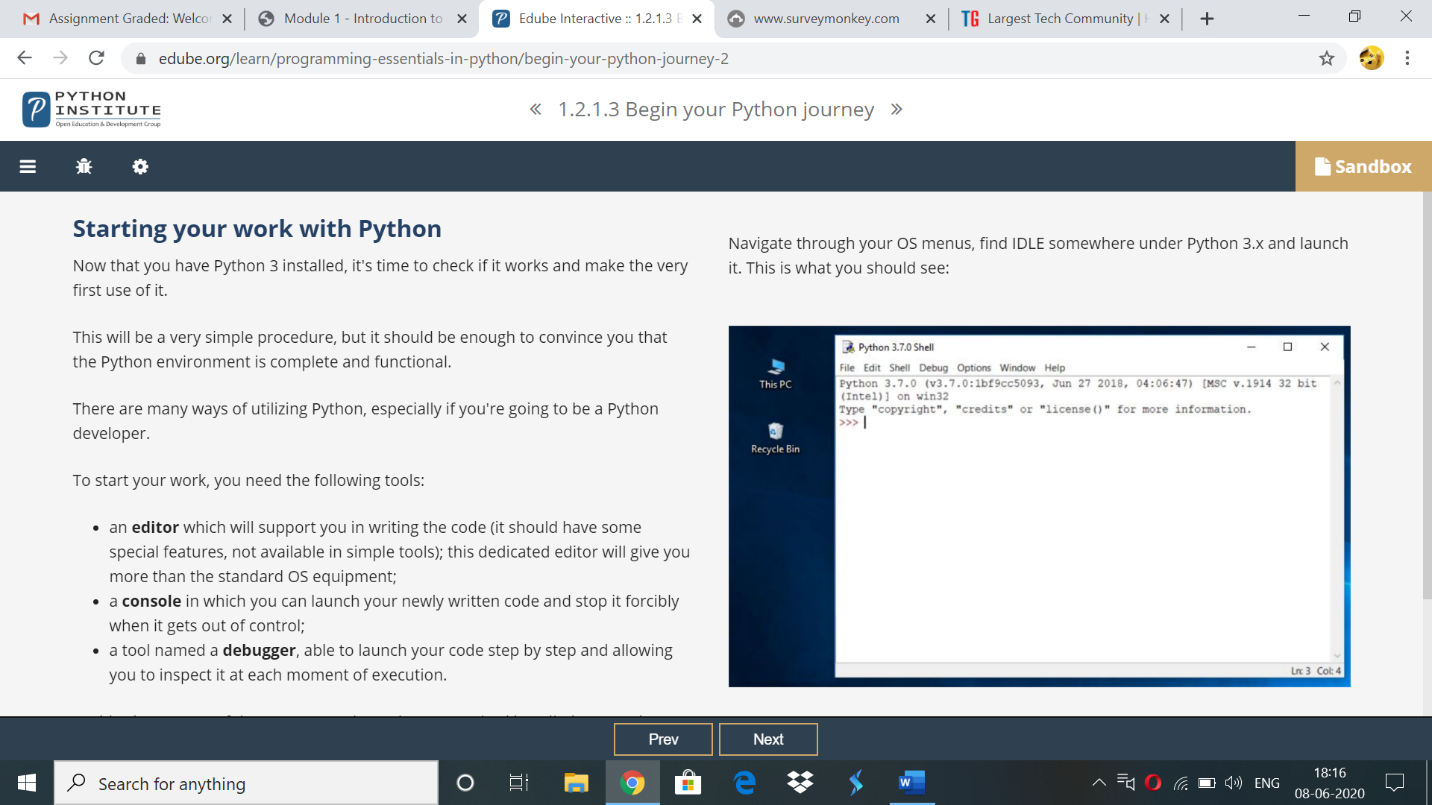
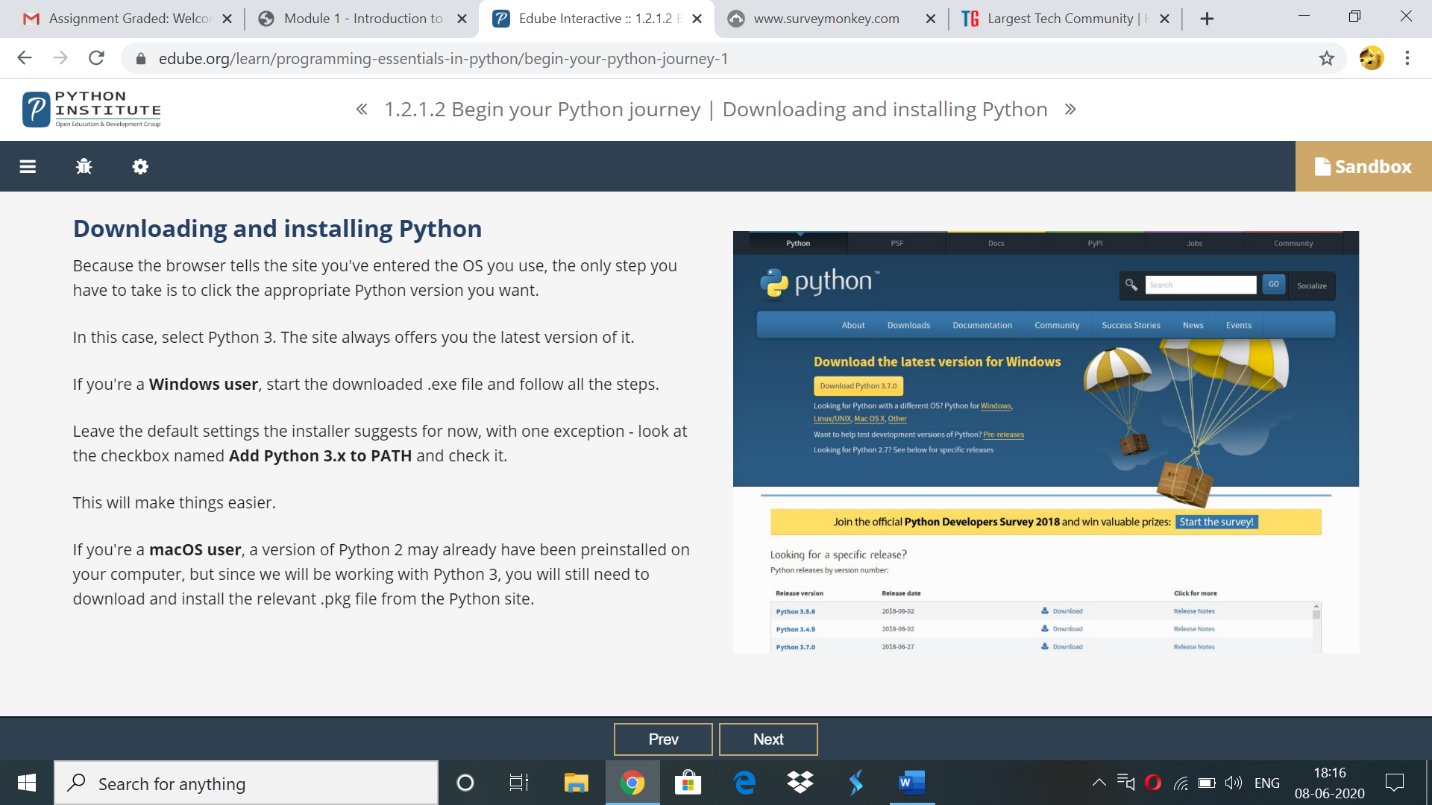
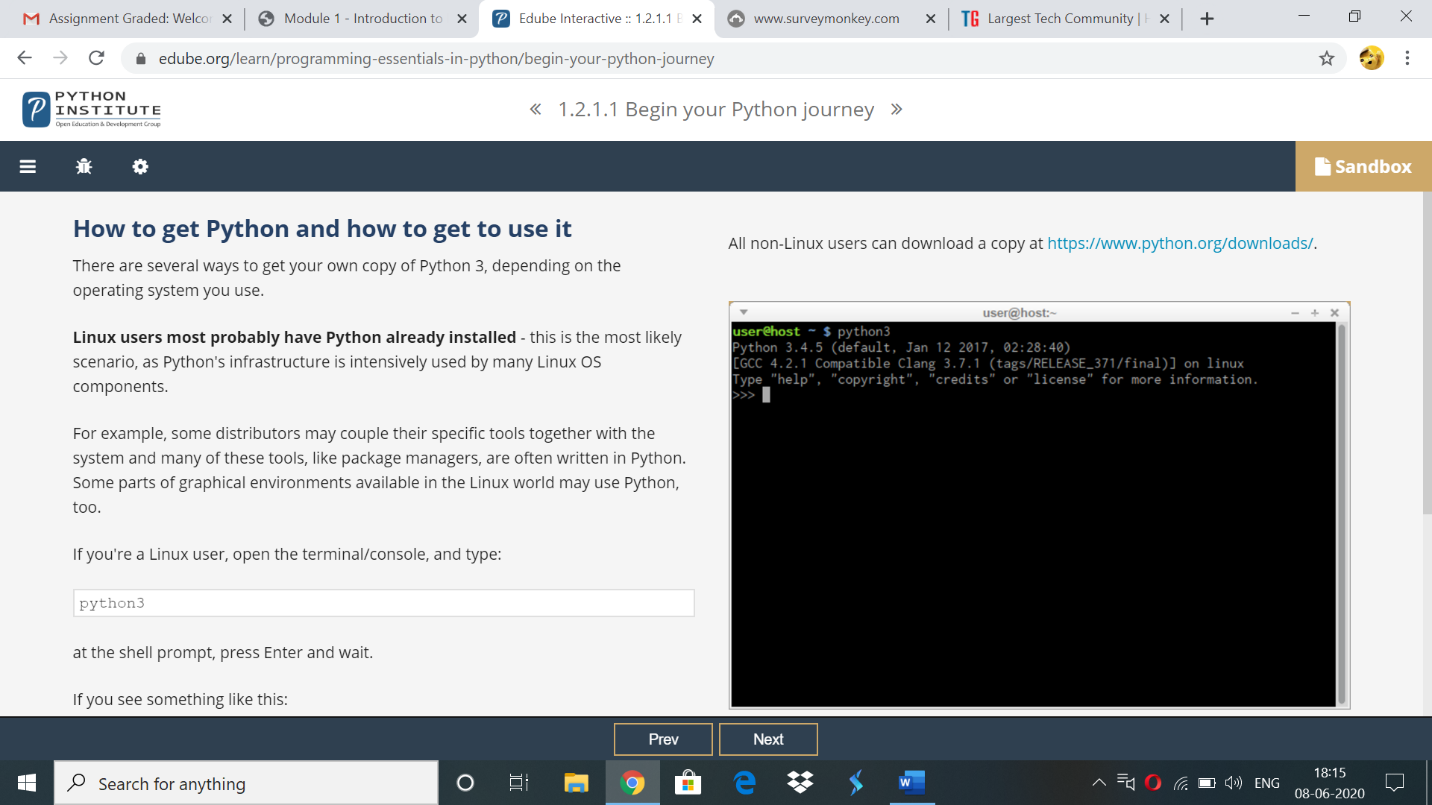
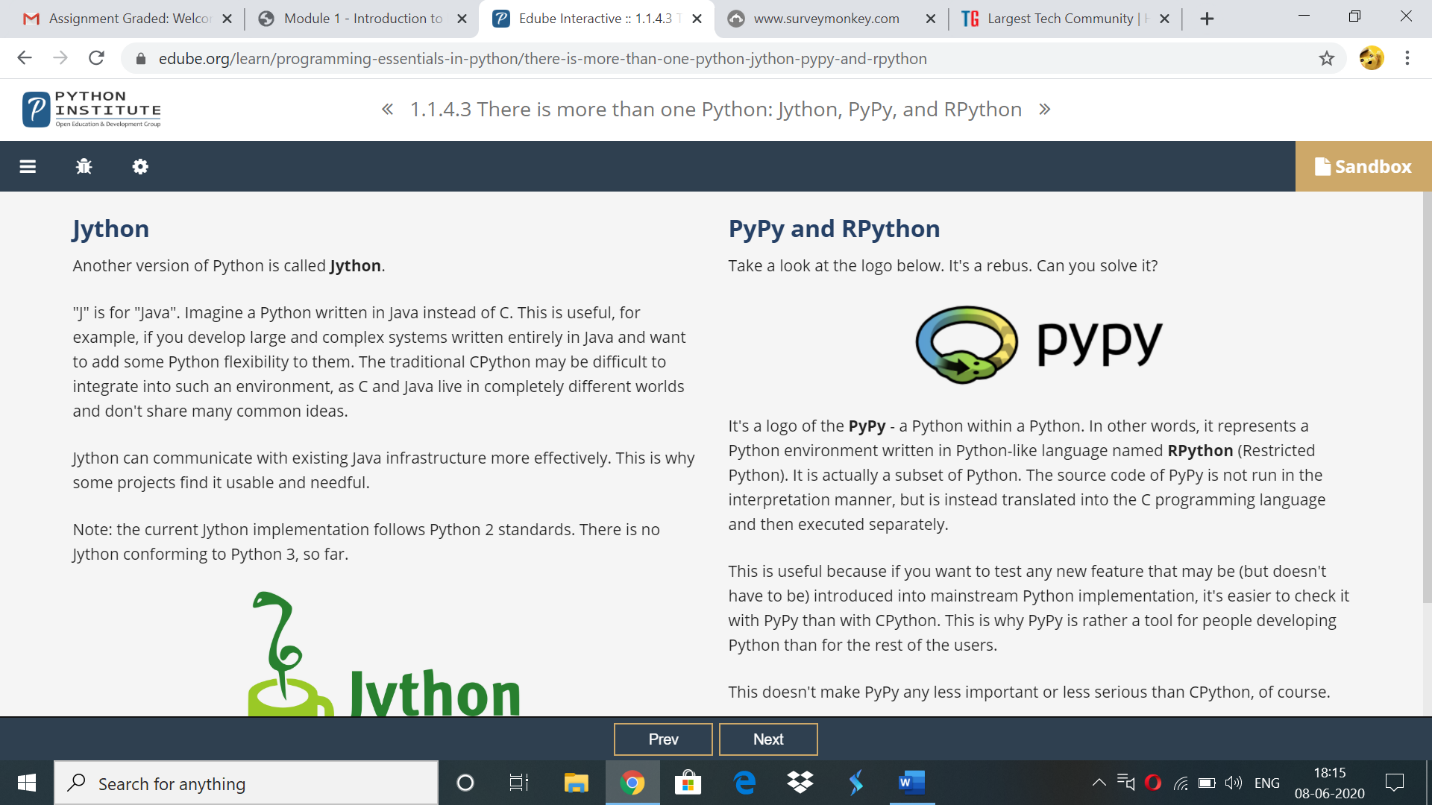
**IA MARKS DETAILS:**

****

**Online Certification Details:**

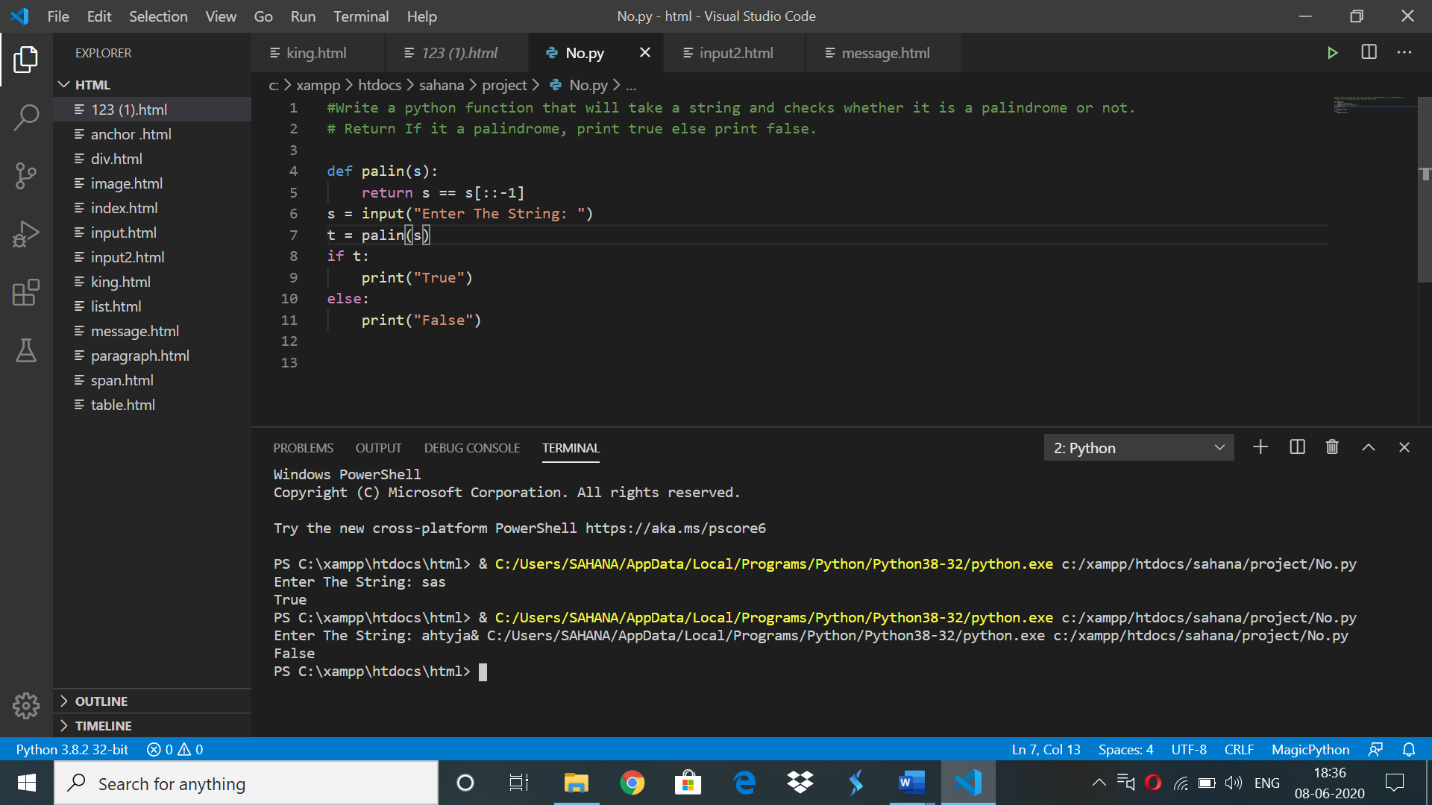
****

Topics covered:



**Online coding:**

**1)**

****

**2)**C Program to Generate All the Set Partitions of n Numbers Beginning from 1 and so on

#include <stdio.h>

#include <stdlib.h>

typedef struct {

int first;

int n;

int level;

} Call;

void print(int n, int \* a) {

int i ;

for (i = 0; i <= n; i++) {

printf("%d", a[i]);

}

printf("\n");

}

void integerPartition(int n, int \* a){

int first;

int i;

int top = 0;

int level = 0;

Call \* stack = (Call \* ) malloc (sizeof(Call) \* 1000);

stack[0].first = -1;

stack[0].n = n;

stack[0].level = level;

while (top >= 0){

first = stack[top].first;

n = stack[top].n;

level = stack[top].level;

if (n >= 1) {

if (first == - 1) {

a[level] = n;

print(level, a);

first = (level == 0) ? 1 : a[level-1];

i = first;

} else {

i = first;

i++;

}

if (i <= n / 2) {

a[level] = i;

stack[top].first = i;

top++;

stack[top].first = -1;

stack[top].n = n - i;

stack[top].level = level + 1;

} else {

top--;

}

} else {

top --;

}

}

}

int main(){

int N = 1;

int \* a = (int \* ) malloc(sizeof(int) \* N);

int i;

printf("\nEnter a number N to generate all set partition from 1 to N: ");

scanf("%d", &N);

for ( i = 1; i <= N; i++)

{

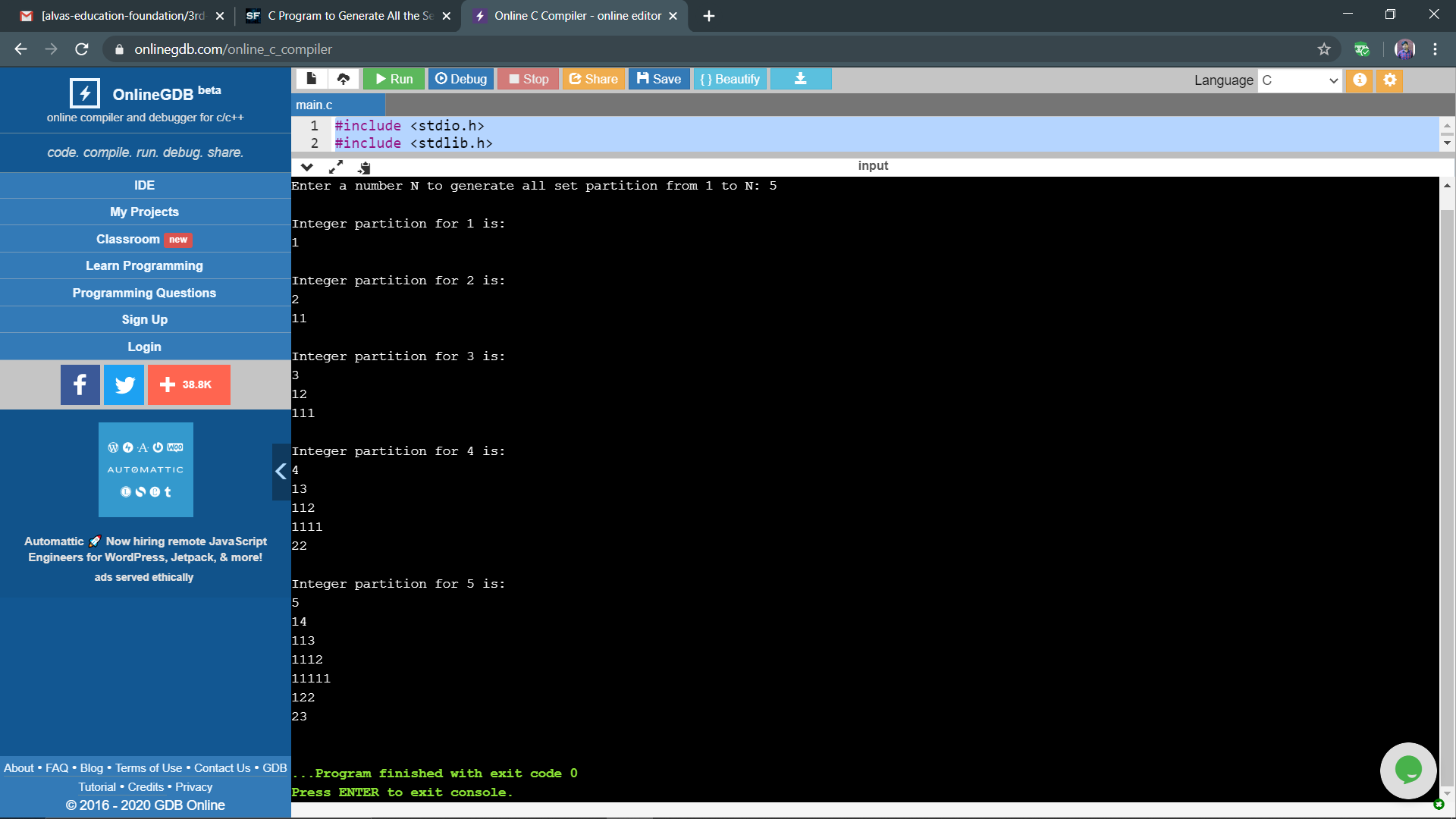
printf("\nInteger partition for %d is: \n", i);

integerPartition (i, a);

}

return(0);

}

OUTPUT: 

**3)** C++ program to check whether a number

#include <bits/stdc++.h>

using namespace std;

// Function to check whether a number

// can be represented by the difference

// of two squares

bool difSquare(int n)

{

// Checking if n % 4 = 2 or not

if (n % 4 != 2) {

return true;

}

return false;

}

int main()

{

int n;

std::cout<<"enter the number: ";

std::cin>>n;

if (difSquare(n)) {

cout << "Yes\n";

}

else {

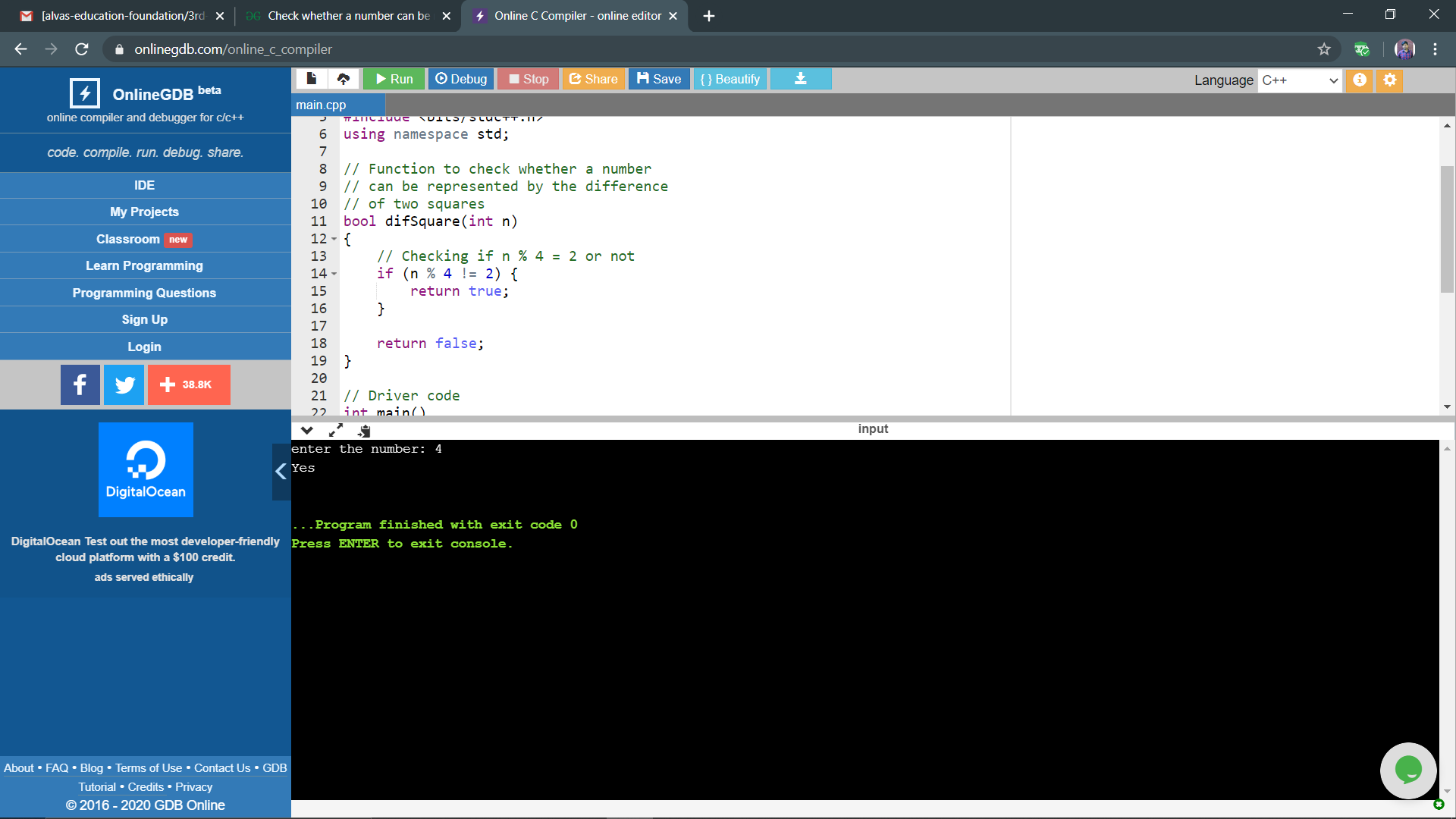
cout << "No\n";

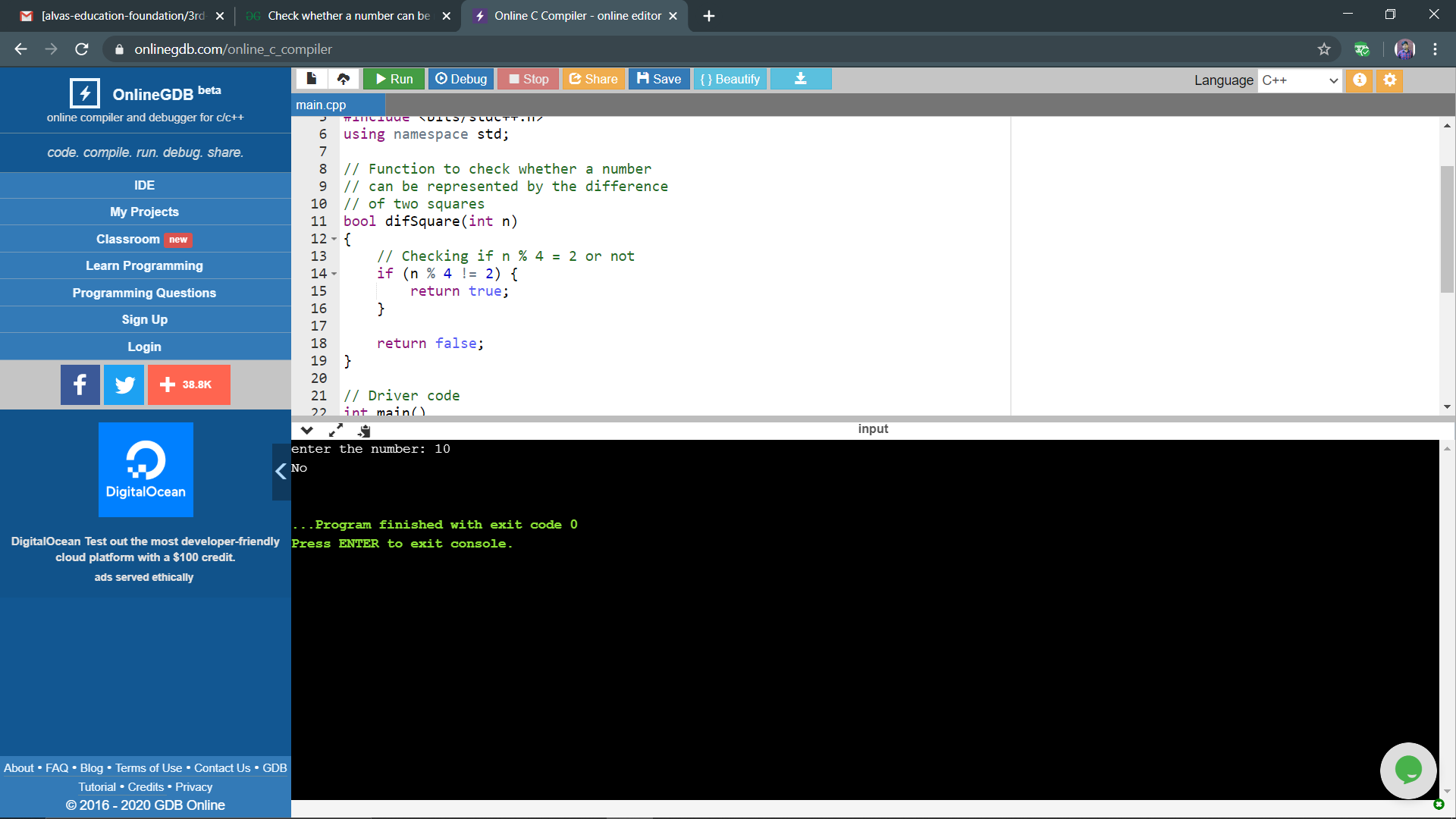
}

return 0;

}

**OUTPUT:**

****

****