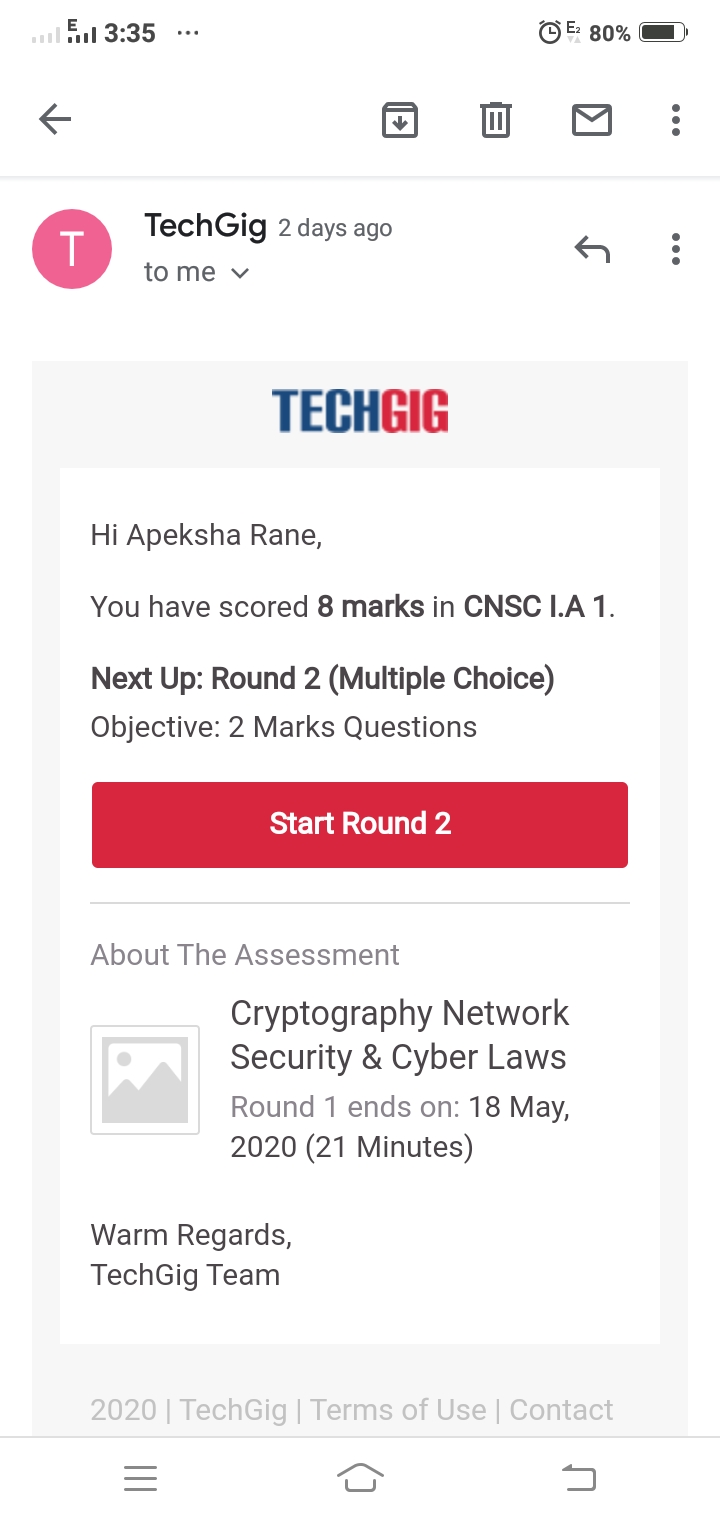
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **18-05-2020** | | | | | **Name:** | **Apeksha Rane** | |
| **Sem & Sec** | **A** | | | | | **USN:** | **4AL17CS010** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **CRYPTOGRAPHY NETWORK SECURITY AND CYBER LAW** | | | | | | |
| **Max. Marks** | | **60** | | **Score** | | | **13** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **INTRODUCTION TO ETHICAL HACKING** | | | | | | | |
| **Certificate Provider** | | | Great learning  Academy | | **Duration** | | | 2hours |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**1**.** Using methods charAt() & length() of String class, write a program to print the frequency of each character in a string.  2. Write down a java program to print even and odd numbers series respectively from two threads: t1 and t2 synchronizing on a shared object.  3. .Write a C Program to check whether two strings are anagram or not. | | | | | | | | |
| **Status: Done** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **YES** | | | |
| **If yes Repository name** | | | | | <https://github.com/Apeksha12appu/18-5-2020-certification-course> | | | |
| **Uploaded the report in slack** | | | | | **YES** | | | |

Online Test Details:

Subject:-CRYPTOGRAPHY NETWORK SECURITY AND CYBER LAW

Marks:13 out of 60



Certification Course Details:

**PYTHON COURSE IN YOUTUBE:**

Today i have learnt :-

1. introduction to python program.

2 installation to python.

3. variables in python.

4. List in python.

5. tuple.

Video link: <https://www.youtube.com/watch?v=QXeEoD0pB3E&list=PLsyeobzWxl7poL9JTVyndKe62ieoN-MZ3>

**Todays certification on Introduction to full stack development**

**About this course**

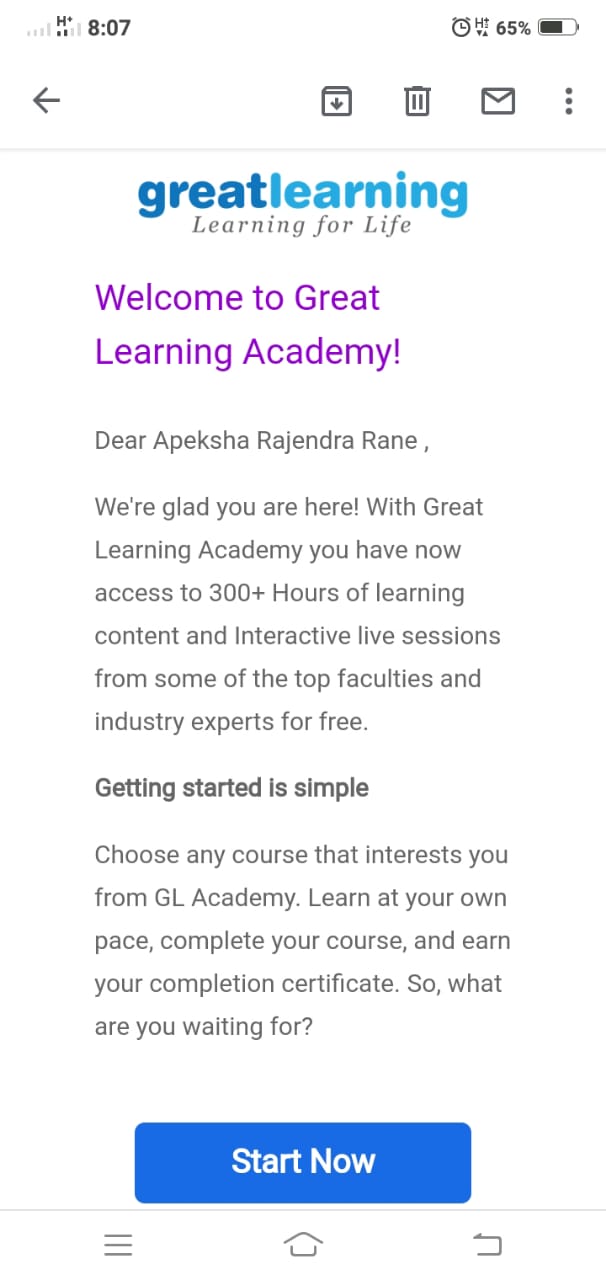
This course will introduce you to frameworks that will help you to become a full stack developer. You will be introduced to HTML basics such as how to create and deploy forms, tables and also learn about different HTML attributes. We will gain an understanding of Cascading Style Sheets which is a language used to describe how to visually present a page that you have written in HTML. You will also learn about keywords such as pseudo-classes that help to achieve event based changes such as changing a button's color when the mouse pointer hovers over it. You will also understand box models and how HTML elements can be modeled in a browser engine.

**Skills covered**

Html

CSS

Front End Development



Coding Challenges Details:

1. Using methods charAt() & length() of String class, write a program to print the

frequency of each character in a string.

“Hello friend”

Output should be

-: 1

d: 1

e: 2

f: 1

(continued for all character in the string)

import java.util.\*;

import java.lang.\*;

import java.io.\*;

class friend {

static final int MAX\_CHAR = 256;

static void getOccuringChar(String str)

{

int count[] = new int[MAX\_CHAR];

int len = str.length();

for (int i = 0; i < len; i++)

count[str.charAt(i)]++;

char ch[] = new char[str.length()];

for (int i = 0; i < len; i++) {

ch[i] = str.charAt(i);

int find = 0;

for (int j = 0; j <= i; j++) {

if (str.charAt(i) == ch[j])

find++;

}

if (find == 1)

System.out.println(" " +

str.charAt(i) + " is:" + count[str.charAt(i)]);

}

}

public static void main(String[] args)

{

Scanner sc = new Scanner(System.in);

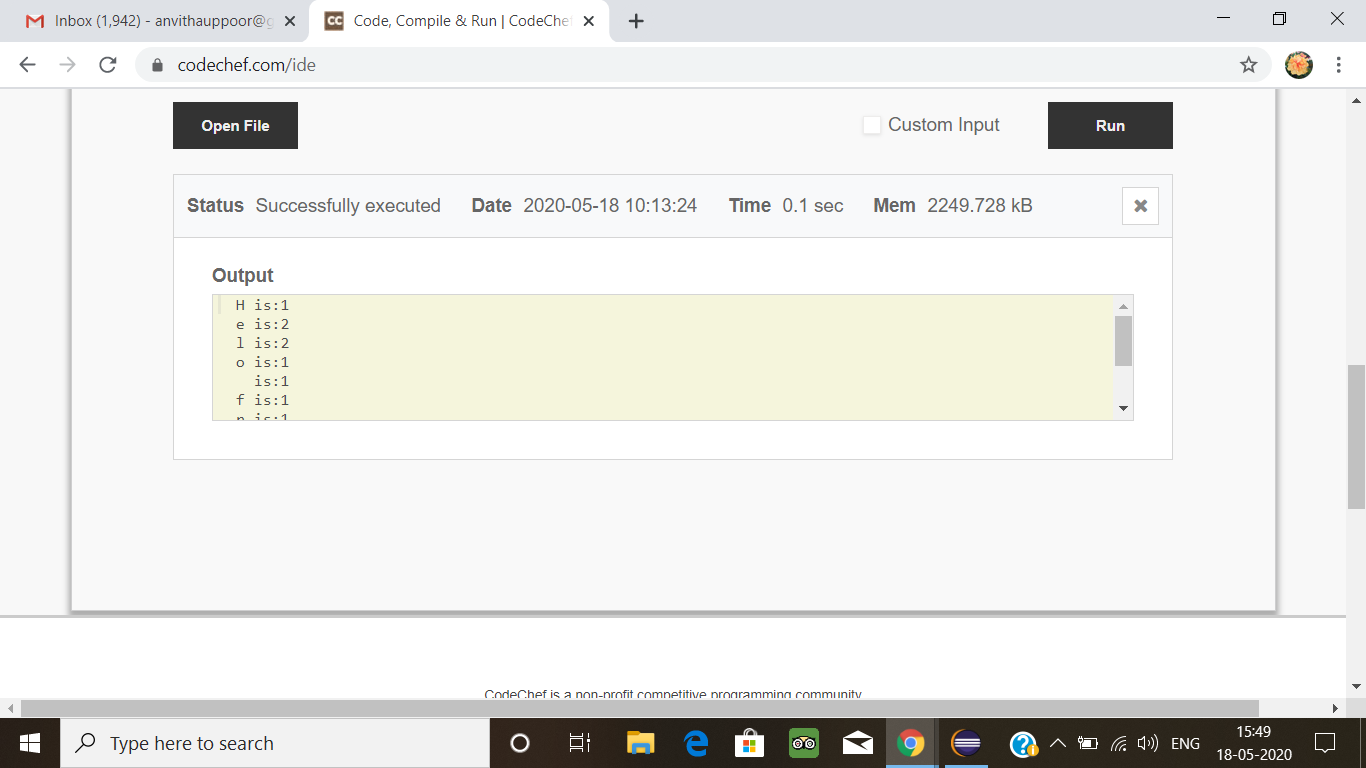
String str = "Hello friend";

getOccuringChar(str);

}

}

**Output:**

****

1. Write down a java program to print even and odd numbers series respectively  
   from two threads: t1 and t2 synchronizing on a shared object

Let t1 print message “ping — >” and t2 print message “,—-pong”.

Take as command line arguments, the following inputs to the program:  
Sleep Interval for thread t1  
Write down a java program to print even and odd numbers series respectively from two  
threads: t1 and t2 synchronizing on a shared object

Let t1 print message “ping — >” and t2 print message “,—-pong”.

Take as command line arguments, the following inputs to the program:

Sleep Interval for thread t1

Sleep Interval for thread t2

Message per cycle

No of cycles

class OddThread extends Thread

{

int limit;

sharedPrinter printer;

public OddThread(int limit, sharedPrinter printer)

{

this.limit = limit;

this.printer = printer;

}

@Override

public void run()

{

int oddNumber = 1;

while (oddNumber <= limit)

{

printer.printOdd(oddNumber);

oddNumber = oddNumber + 2;

}

}

}

class EvenThread extends Thread

{

int limit;

sharedPrinter printer;

public EvenThread(int limit, sharedPrinter printer)

{

this.limit = limit;

this.printer = printer;

}

@Override

public void run()

{

int evenNumber = 2;

while (evenNumber <= limit)

{

printer.printEven(evenNumber);

evenNumber = evenNumber + 2;

}

}

}

class sharedPrinter

{

boolean isOddPrinted = false;

synchronized void printOdd(int number)

{

while (isOddPrinted)

{

try

{

wait();

}

catch (InterruptedException e)

{

e.printStackTrace();

}

}

System.out.println(Thread.currentThread().getName()+" "+number);

isOddPrinted = true;

try

{

Thread.sleep(1000);

}

catch (InterruptedException e)

{

e.printStackTrace();

}

notify();

}

synchronized void printEven(int number)

{

while (! isOddPrinted)

{

try

{

wait();

}

catch (InterruptedException e)

{

e.printStackTrace();

}

}

System.out.println(Thread.currentThread().getName()+" "+number);

isOddPrinted = false;

try

{

Thread.sleep(1000);

}

catch (InterruptedException e)

{

e.printStackTrace();

}

notify();

}

}

public class Main

{

public static void main(String[] args)

{

sharedPrinter printer = new sharedPrinter();

OddThread oddThread = new OddThread(20, printer);

oddThread.setName("—-pong");

EvenThread evenThread = new EvenThread(20, printer);

evenThread.setName("ping — >");

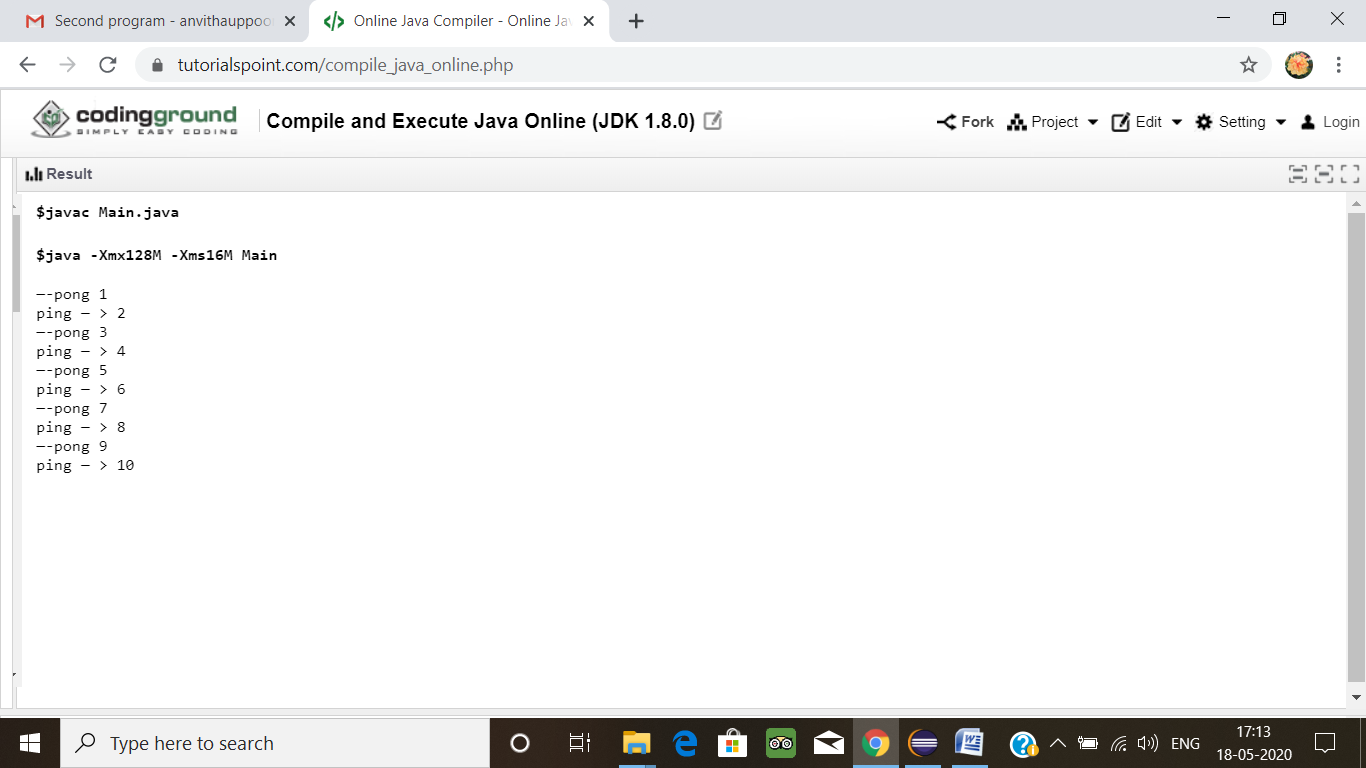
oddThread.start();

evenThread.start();

}

}

**Output:**

****

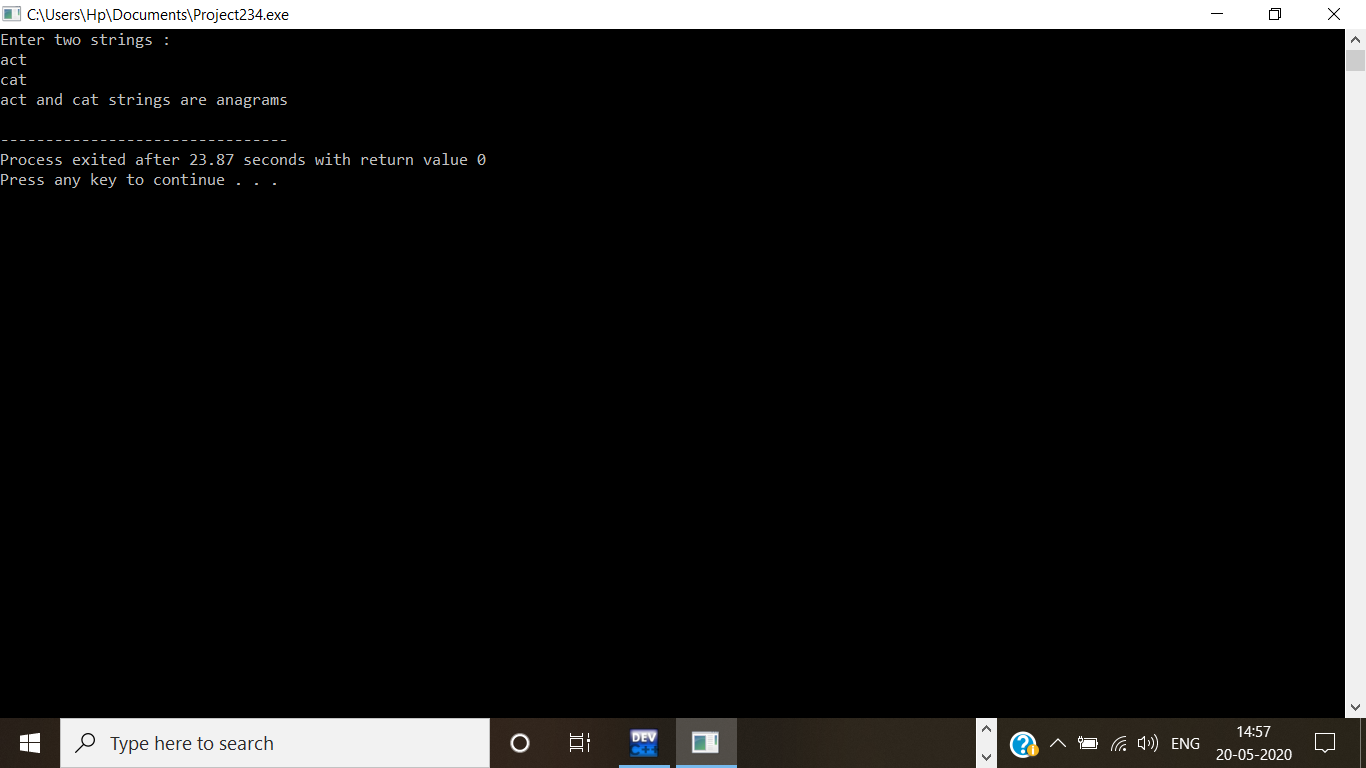
3.Write a C Program to check whether two strings are anagram or not.

Hint: An Anagram of a string is another string that contains same characters, only the order of characters can be different.  
For example, "act" and "cat" are anagram of each other.

#include <stdio.h>  
int check\_anagram(char [], char []);  
int main()  
{  
char a[100], b[100];  
printf("Enter two strings : \n");  
gets(a);  
gets(b);

if (check\_anagram(a, b) == 1)  
printf("%s and %s strings are anagrams\n",a,b);  
else  
printf("%s and %s strings are not anagrams\n");

return 0;  
}  
int check\_anagram(char a[], char b[])  
{  
int first[26] = {0}, second[26] = {0}, c=0;  
while (a[c] != '\0')  
{  
first[a[c]-'a']++;  
c++;  
}  
c = 0;  
while (b[c] != '\0')  
{  
second[b[c]-'a']++;  
c++;  
}  
for (c = 0; c < 26; c++)  
{  
if (first[c] != second[c])  
return 0;  
}  
return 1;  
}

**Output:** ****