

Name: Himanshu Rana Roll no: 1121063 (07) Subj name: Information Security
and its laws.

Ans)

objective: A google account is the key to accessing all of google's products and services.

a) check google privacy policies

Step1 :- Log to your google account

Step2:- Go to google Privacy Policies and check the policies associated with it.

Step3:- Following are some google Privacy Policy

1. Privacy reminders from google.
2. Third party sites and apps with access to your account.
3. See, control and delete the information in your google account.
4. Change your Privacy settings.
5. Download your data.
6. Make your account more secure
7. Use google Smart lock.

b) check your account recovery

Step1: Login to your google account

Step2: Go to your security option

Step3: Click on recovery phone and recovery email one by one

Step4: First you have to sign in again to your google account for verification.

Step5: Now you can recover your account by adding phones and email both by one.

Step6: Account recovered successfully.

↳ check your account security.

Step 1: - Login to your google account

Step 2: Go to help option where you find tips related to

your google account.

Step 3: Following are the options comes under in help:

1. Help with common issues like control and recovery of data.
2. Guiding steps for adding privacy, account protection and binding your device.
3. Discuss your problems related to your google account with other people who use the service as you.
4. You can report your issues and get solution from that.
5. You can also give feedback to your google service.

Name: Himanu Rana Rollno.: 121063 (02)
Subj name: Information security and cyber law.

Q) write a program to implement OTP

#include <stdio.h>
#include <string.h>

#include <ctype.h>
#include <string.h>

int main()

```
char plantxt[100], otp[100];
printf ("Enter plain text \n");
fflush (stdin);
gets (plantxt), stdin );
printf ("Enter otp of length 10d ", stdlen(plantxt));
fflush (stdin);
gets (otp, size of (otp), stdin);
for (int i=0; i < stdlen(plantxt); i++)
{
    if (i >= otp[i])
        otp[i] = toupper(otp[i]);
    else
        otp[i] = (otp[i] - 'A' + plantxt[i] - 'A') % 26;
}
```

$$\text{if } (\text{plantxt}[i] + (\text{otp}[i] - 'A')) \% \text{plantxt}[i] \\ = \text{plantxt}[i] + (\text{otp}[i] - 'A') ;$$

else (plantx4Li) = plantx4Li; }
3

Print ("cypho +t+ it's\in", plantx4);

return plantx4; }
else if (plantx4Li == plantx4Li2) {
 print ("Cypho is identical to Li");
 return 0;

else if (plantx4Li == plantx4Li3) {
 print ("Cypho is identical to Li3");
 return 0;

else if (plantx4Li == plantx4Li4) {
 print ("Cypho is identical to Li4");
 return 0;

cout << "Cypho is not identical to Li";

cout << "Cypho is not identical to Li3";

cout << "Cypho is not identical to Li4";

cout << "Cypho is not identical to Li2";

else { cout << "Cypho is not identical to Li";
 cout << "Cypho is not identical to Li3";
 cout << "Cypho is not identical to Li4";
 cout << "Cypho is not identical to Li2";
}

cout << "Cypho is not identical to Li";
cout << "Cypho is not identical to Li3";
cout << "Cypho is not identical to Li4";
cout << "Cypho is not identical to Li2";
cout << "Cypho is not identical to Li";

Cypho is not identical to Li";
cout << "Cypho is not identical to Li3";
cout << "Cypho is not identical to Li4";
cout << "Cypho is not identical to Li2";
cout << "Cypho is not identical to Li";

Cypho is not identical to Li";
cout << "Cypho is not identical to Li3";
cout << "Cypho is not identical to Li4";
cout << "Cypho is not identical to Li2";
cout << "Cypho is not identical to Li";

Name: Himanshu Rana Rollno - 1121063(07)
Subj name: Information security and cyber law

Ques Encryption using caesar cipher.

def encrypt (String),

cipher = ""

for char in string:

if char == " ":

cipher = cipher + char

~~char - cipher~~

elif char. issupper ():

cipher = cipher + chr((ord (char) + 3 - 65) % 26

+ 65);

else:

cipher = cipher + chr((ord (char) + 3 - 97) % 26
+ 97);

return cipher

text = "Attack from North",

Print ("After encryption: " encrypt (text)).

Decryption using caesar cipher:

def decrypt (string):

plain = "

for char in string:

if char == " ":

plain = plain + char

elif char. issupper ():

plain = plain + chr ((ord (char) - 3 - 65) % 26 + 65)

else:

plain = plain + chr ((ord (char) - 3 - 97) % 26
+ 97)

return plain

← `text = "`

← Print ("After decryption: ", decrypt (hex+)).

Qimansh

0244

00000000000000000000000000000000

00000000000000000000000000000000

0000

00000000000000000000000000000000

0000

00000000000000000000000000000000

00000000000000000000000000000000

00000000000000000000000000000000

00000000000000000000000000000000

00000000000000000000000000000000

00000000000000000000000000000000

Name: Himanshu Rana Roll No - 121063 (07)

Subject: Information security & and Cyber law.

MCA

- ① Asymmetric key encryption with sender's public key
- ② Spyware
- ③ An authentication of an electronic record
- 4) Cyber security
- 5) only on ASCII coded data
- 6) All
- 7) hash value
- 8) The identification of the character is changed while its position remains unchanged.
- 9) To make even no. of letters.
- 10) Total length of word.