

# Information Security and Cyberlaw

Date.....

(PBC-601)

Name: Himaneshu Singh Rana (BCA Section 'C')

Roll NO:- 1121064 (08)

Ans 4:-

```
import math, random  
def generate OTP():
```

```
    digits = "0123456789"
```

```
    OTP = ""
```

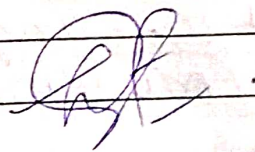
```
    for i in range(4):
```

```
        OTP += digits [math.floor(random.  
            random () * 10)]
```

```
    return OTP
```

```
if __name__ == "__main__":
```

```
    print("OTP of 4 digits:", generate OTP())
```





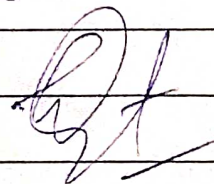
Ans 3:-

```

def generateKey (String, key):
    key = list (key)
    if len (String) == len (key):
        return (key)
    else:
        for i in range (len (String) - len (key)):
            key.append (key[i % len (key)])
        return (" ".join (key))
def encryption (String, key):
    Cipher_text = []
    for i in range (len (String)):
        x = (ord (String[i]) + ord (Key[i])) % 26
        x += ord ('A')
        Cipher_text.append (chr (x))
    return (" ".join (Cipher_text))
def decryption (Cipher_text, key):
    org_text = []
    for i in range (len (Cipher_text)):
        x = (ord (Cipher_text[i]) - ord (Key[i]) + 26) % 26
        x += ord ('A')
        org_text.append (chr (x))
    return (" ".join (org_text))

if __name__ == "__main__":
    String = "Cryptography"
    Key = "Monarchy"
    Key = generateKey (String, Key)

```



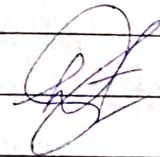


Date.....

CipherText = encryption (string, key)

Print ("CipherText: ", CipherText)

Print ("original (Decrypted text: ", decryption  
(CipherText, key)





Ans 5:- def encrypt (string):  
 cipher = ""  
 for char in string:  
 if char == " , ,":  
 cipher = cipher + char  
 elif char.isupper():  
 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".

\* Decryption using Ceaser Cipher.

def decrypt (string):  
 plain = ""  
 for char in string: