NAME - BHARAT GURUNG
BCA - A, 6th SEM
1121057, class roll - 34

Information Security and cyberlaws

MCQ.

1.) b) Asymmetric key encyption with senders public key

2) c) Spyware

3) d) An authentication of an electronic record.

4) c) Cyber laws

5) a) Only on alphanumeric

6) d) Idea is same title is different

7) a) Hash Value

8) a) The position of the character is changed in spite of its identity

9) d) both b and c

10) d) None

NAME-BHARAT GURUNG
BCA A 6th SEM
112087, Classroll-34

Information Security and Cyber laws.

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 CipherText (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))
```

NAME - BHARAT GURUNG
BCA - A, 6th sem
11210374, class roll - 34
Information Security and Cyber laws

Ans4.

```
# import Library
import math, random

# function to generate OTP

def generate OTP();

    # Declare a digits variable
    # which stores all digits

    digits = "0123456789"

    OTP = " "

# length of password can be changed
# by changing value in range
    for i in range (4):

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

    return OTP

# Driver code

if __name__ = "__main__":
    print ("OTP of 4 digits :", generateOTP())
```

Gurung

NAME - BHARAT GURUNG
BCA-A, 6th SEM
11R1037, class roll - 54

Information Security and Cyber laws

Ans 5) Encryption using Caesar Cipher:

```
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"
print("After Encyption:", encrypt(text))
```

Decryption using Caesar cipher

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

```
if char in string:
    if char == ( ):
    if char == ' ':
    plain = plain + char
    elif char.isupper():
    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(text))
```