END SEMESTER PRACTICAL.

33
Class Roll no.

Name :- Rubiya Rawat.

father's Name :- Late M.P.S Rawat

University Roll no :- 1121114.

Course :- BCA "B"

Semester :- 6

Paper Name :- Information Security Lab.

Paper Code :- PBC - 601.

Type of Paper :- Regular

Date of Examination :- 15th June' 2021.

MCQ.

Q1 a) Symmetric Key encryption with reciever public Key.

Q2 c) Spyware.

Q3 c) An authentication of an electronic record.

Q4 d) None.

Q5 a) Only on alphanumeric

Q6 c) All.

Q7 a) hash value.

Q8 b) The identity of character is changed while its position remains unchanged.

Q9 b) to make even no. of letters.

Q10 a) Total length of word.

Rubiya.

Q5 **Caeser Cipher**

```
print ("Encryption")
def(encrypt (text, s):
 result = ""
 for i in range(len(text)):
    char = text[i]
    if(char.isupper()):
       result = result + chr((ord(char)+s-65)%26+65)
    else:
       result = result + chr((ord(char)+s-97)%26+97).

  return result

text = "Attack from North"
s=3
print("Plain text:", text)
print ("Encrypted text:", encrypt(text, s))
print ("Decryption")
def decrypt(text, s):
   result = ""
   for i in range(len(text)):
      char = text[i]
      if(char.isupper()):
         result = result+chr((ord(char)-s-65)%26+65)
      else:
         result = result + chr((ord(char)-s-97)%26+97]
    return result
text = encrypt(text, s)
 s=3
print ("Decrypted text:", decrypt(text,s)).
```

Rubiya

Que 4.

```
import math, random
def generateOTP( )
    digits = "234207852"
    OTP = ""
    for i in range(4):
        OTP+= digits[math.floor(random.random()*10)]
    return OTP
if _name_ == "_main_":
    print("OTP of 4 digits is :", generate OTP())
```

Que 3

```
def generate(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):
    encrypt_text=[]
    for i in range(len(string)):
        x = (ord(string[i] + ord(Key[i])) % 26
        ~~x [illegible]~~
        x += ord('A')
        encrypt_text.append(chr(x))
    return ("".join(encrypt_text))
```

Rukiya

```
def decryption(encrypt_text, key):
    orig_text = "cryptography"
    for i in range(len(encrypt_text)):
    x = (ord(encrypt_text[i]) - ord(key[i]+26) % 26.
    x += ord('A').

    orig_text.append(car(x))
  return ("".join(orig_text)
if _name_ == "__main__":
   string = "cryptography"
   ~~keyword = input ("Enter the keyw~~
   keyword = "monarchy".
   key = generate key(string, keyword)
   encrypt_text = encryption(string, key).
   print(" Encrypted text", encrypt_text)
   print ("Decrypted text", decryption(encrypt_key, key])
```

Rubiya.