

Name :- Amit Dobhal
Course :- BCA 'C'
Sem = Vth
Subject :- Info Security
Roll No :- 1121015

MCQ

- 1 :- Asymmetric key encryption with Sender's public key
- 2 :- Spyware
- 3 :- An authentication of an electric record
- 4 :- Cyber laws
- 5 :- Only an alphanumeric
- 6 :- Idea in same title is different
- 7 :- Hash value
- 8 :- The identity character is changed with its position.
- 9 :- both b and c.
- 10 :- Possibility of replacement

Name :- Amit Dobhal
Course :- BCA 'C'
Roll No :- 1121015
Subject :- Info. Security.

Ques 1 :- 3 Security aspects of
google account.

1 Step :- Go to Security checkup
to get personalized security

2 Step :- Verification helps prevent
a hacker from getting into your
account. even if they steal your
password. To avoid common
phishing techniques associated
with text message code.

Update your software if your
browser, operating system or app
are out of date.

3 Step :- Use unique & strong
password. make sure
to create a unique and
strong password.

Name :- / Amit Dobtal
Course :- / BCA VIth
Sub :- / Info. Security
Roll :- / 1122015.

Qus 2 :-

See control and delete
info in your google account.

Step 1 :- Login your account

Step 2 :- Go to dashboard

Step 3 :- Now you can see someone
popular service like gmail
activity history etc.

Step 4 :- You have also more ways
to control your data like
Security check up.

Step 5 :- Now make some change
to your google service

Step 6 :- Change done
Successfully.

Name :- Amit Dobhal

Course :- BCA 'C'

University Roll No :- 1121015

Subject :- Information Security

Ques 3 :-

C++ Program Vigenere cipher

```
#include <iostream>
#include <string.h>
using namespace std;
int main() {
    char msg[] = "Cryptography"
    char key[] = "Monarchy"
    int msgLen = strlen(msg), keyLen =
        strlen(key), i, j;
    char newkey[msgLen], encryptedMsg[msgLen],
        decryptedMsg[msgLen];
    for (i = 0; j = 0; i < msgLen; ++i, ++j) {
        if (j == keyLen)
            j = 0;
        newkey[i] = key[j];
    }
    newkey[i] = '\0';
    for (i = 0; i < msgLen; ++i)
        encryptedMsg[i] = ((msg[i] + newkey[i] % 26
            + 'A')) % 26;
    encryptedMsg[i] = '\0';
    for (i = 0; i < msgLen; ++i)
        decryptedMsg[i] = (((encryptedMsg[i] -
            newkey[i]) + 26) % 26) + 'A';
}
```



```
decryptedMsg[i] = '\0';  
cout << "Original Message: " << msg;  
cout << "\n key: " << key;  
cout << "\n new Generated key: " << newkey;  
cout << "\n Encrypted message: " << encryptMsg;  
cout << "\n Decrypted message: " << decryptedMsg;  
return 0;  
}
```


Name:- Amit Dobhal
University RollNo:- 1121015
Course:- BCA
Subject :- Information Security
Subject code:-

Ques 4 -

Python.

4 digit Numeric OTP

```
#import library
import math, random

# Function to generate OTP
def generateOTP():
    # 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())
```

Output

OTP of 4 Digit : 3211

Name :- Amit Dobhal

RollNo :- 1121015

Subject :- BCA 'C' Sem = 6th

Course :-
Subject :- Information Security.

Ques 5 :- Encryption using caesar cipher

```
def encrypt (String):
```

```
    cipher = ""
```

```
    for char in string:
```

```
        if char == ' ':
```

```
            cipher = cipher + char
```

```
        elif char.isupper():
```

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

```
        else:
```

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

```
    return cipher
```

```
text = "Attack from North"
```

```
print ("after Encryption", encrypt(text))
```

decryption

```
def decrypt (String)
```

```
    Plain = ""
```

```
    for char in string:
```

```
        if char == ' ':
```

```
            Plain = Plain + char
```

```
        elif char.isupper():
```



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

```
else :
```

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

```
return plain
```

```
text = ""
```

```
printf("after decryption" decrypt  
(text))
```