Name: Devashree Maharjan
Course: BCA   Sem: 6th   Sec: 'A'
University roll no: 1121041

Multiple Choice Questions:

1. Asymmetric key encryption with sender public key.
2. Spyware
3. An authentication of an electronic record
4. Cyber laws
5. Only on alphanumeric
6. Idea is same title is different
7. Hash values
8. The identity of the character is changed while its position remains unchanged.
9. both b and c
10. Possibility of replacements

Devashree

Answer 1: 3 security aspects of the Google account:-

(i.) Do a security check up

Go to security checkup to get personalized security recommendations for your google account including

- Turn on 2-step verification
- Turn on screen locks
- Update account recovery options.

(ii.) Update your software

If your browser, operating system, on apps are out of date the software might not be safe from hackers.

- Update your browser, apps and operating system.

(iii.) Use unique, strong passwords

Its risky to use the same passwords on multiple sites. If password for one site is hacked it could be used to get into your accounts for multiple sites.

- Manage your password
- Protect your password from hackers.

Devashree

Answer 4:

```
#import library
  import math.random
#function to generate OTP
  def generateOTP();
     #Declare a digit 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
                              (1 * 10))]
     return OTP
   # Driver code
   if __name__ ="__main__":
    print(“OTP of 4 digits: ”, generateOTP())
```

Devashree

Answer 5:

Encryption using Caeser Cipher:-

Devashree

```
def encrypt (string):
cipher = "  "
for char in string:
  if char = = '  ':
    cipher = cipher + char
  elif char. isupper( ):
    cipher = cipher + chr ((cord (char) + 3 - 65)
    % 26 + 65)
  else:
    cipher = cipher + chr ((cord (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.isupper( ):
plain = plain + chr ((ord )-3-65) % 26 +65)
else :
  plain = plain + chr ( cord (char ) - 3 - 97) % 26 +97)
return plain
← text= "        "
← print(" after decryption: ", decrypt (text))
```