Assignment1

**OpenSSL & RSA**

1. **Show your setup environment and you have installed openssl(on any environment like WSL,Mac OS, Linux, etc) (2 Points)**

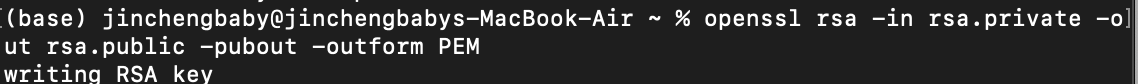
My system is Mac os and openssl version is 1.1.1. k.



**2. Generate private and public key (2 Points)**

Text

Description automatically generated



1. Pick a file, encrypt it with the public key **(3 Points)**

[Your response goes below.]

Text

Description automatically generated

1. Decrypt the encrypted file with the private key **(3 Points)**

        [Your response goes below.]



**II. Caesar Encryption (10 Points)**

1. Suppose we have Shift Key = 13, and text = “SECURITY IS IMPORTANT”

What would be the encrypted message? (2 Points)

Note, the shift is right shift, that is key = 1, ‘a’ -> ‘b’

[Your response goes below.]

The answer would be “FRPHEVGL VF VZCBEGNAG”

1. Now, given the text "SECURITY IS IMPORTANT" and the encrypted message you got from Step 2. Can you write a brute force function that returns the key?

You need to write the function code and show that it passes the test case "SECURITY IS IMPORTANT". You will also need to provide a screenshot for this step.(4 Points)

[Your response goes below.]

Text

Description automatically generated with medium confidence

1. What is the time and space complexity of the hacking (2 Points)

[Your response goes below.]

Set the length of message to be N , the letter length would always be 26, the time complexity is O(26 \* N) =🡺 The time complexity would be O(N)

The space complexity would also be O(N)

1. If the text size is large, could you find a better way to do the hacking or approach the problem? Explanation is enough. No need to post the code screenshot. (2 Points)

[Your response goes below.]

Using the dictionary to store the 26 shift of alphabet which would increase the loop up time complexity to be O(1), it might increase the overall speed.

**III. Diy: make your own cipher (Open Design Problem) 10 Points**

1. Completeness of your codes that contains at least encrypt and decrypt functionalities **(3 Points)**

monoalpha\_cipher = {

'a': 'm',

'b': 'n',

'c': 'b',

'd': 'v',

'e': 'c',

'f': 'x',

'g': 'z',

'h': 'a',

'i': 's',

'j': 'd',

'k': 'f',

'l': 'g',

'm': 'h',

'n': 'j',

'o': 'k',

'p': 'l',

'q': 'p',

'r': 'o',

's': 'i',

't': 'u',

'u': 'y',

'v': 't',

'w': 'r',

'x': 'e',

'y': 'w',

'z': 'q',

' ': ' ',

}

inverse\_monoalpha\_cipher = {}

for key, value in monoalpha\_cipher.items():

inverse\_monoalpha\_cipher[value] = key

def encrypt(message):

encrypted\_message = []

for letter in message:

encrypted\_message.append(monoalpha\_cipher.get(letter, letter))

return ''.join(encrypted\_message)

def decrypt(encrypted\_message):

decrypted\_message = []

for letter in encrypted\_message:

decrypted\_message.append( inverse\_monoalpha\_cipher.get(letter, letter))

return ''.join( decrypted\_message )

1. Your explanation of your design how to approach the problem  **(4 Points)**

[Your response goes below.]

A monoalphabetic cipher uses fixed substitution over the entire message.

We can build a monoalphabetic cipher using a Python dictionary, from the above dic, is the originl encrypt key and we need to reverse the key to have the one dicionary used for decryption. Then we can use the two dictionary to encrypt and decrypt using simple iteration through the message.

1. Write and pass some test cases. **(3 Points)**

****

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated