RSA ASSIGNMENT

The aim of the assignment is to make sure you understand and get familiarized with the concepts of the RSA Algorithm. By the end of the assignment, you should have

- Understand how the RSA algorithm works.
- Write code to generate public and private keys
- Write code to encrypt the plain text using the public key
- Write code to decrypt the data using the private key

Pre-requisite:

Reading: Lookup the *Public and Private Key Encryption* Slides and understand how private key encryption works. Also, read about the RSA algorithm, steps to create a public and private key.

Lab Setup: Partially completed code is provided to you. Open *'RSA.py'* in the python IDE of your choice.

Programming Assignment

RSA is one of the widely used public key encryption algorithm. The algorithm is composed of three key parts:

1) public and private key generation [*key_gen()*]
2) encryption [*encrypt()*]
3) decryption [*decrypt()*].

The above steps are implemented in the given program as *key_gen(), encrypt(), decrypt()* methods respectively. Your task is to implementation these methods. The methods and their respective tasks are listed below:

key_gen()

1) Takes public_key_pair and user_message as input
2) Check if p and q are primes
3) Check if p and q are equal
4) Compute the value of n such that: n= p * q
5) Find the value of totient(t) such that: t = (p-1) * (q-1)
6) Choose integer e such that e and t are co-prime [HINT: Use the defined gcd() method]
7) Choose the value of d such that (p * q) mode n = 1 [find multiplicative inverse using Extended Euclid's Algorithm] [already done for you multiplicative_inverse(e, t)]
8) Generate return the public and private key.

encrypt()

1) Takes generated public_key_pair and user_message as input.
2) Convert each letter to its ASCII value and Encrypt each value using the formula:
   encryption = (message ^ e) mod N
3) return ciphered text array

decrypt():

1) Takes private_key_pair and cipher as input
2) Decrypt the cipher using the formula:
   decryption = (encrypted_message ^ d) mod N
3) Convert ASCII code to characters
4) Return the decrypted_text

Helper Functions:

gcd()

- Find the Greatest Common Divisor (GCD) of two numbers.

  Note: Two numbers are co-prime if their GCD = 1

prime_check():

- Check if a number is prime or not.

The sample expected output is as follows:

```
!!!===== Encryption and Decryption using RSA Algorithm ========!!

 Enter a prime number (greater than 10): 11
 Enter another prime number (Not one you entered above and greater than 10): 13
 Generating Public and Private key....
 t:  120
 e:  63
 d:  127
 Public key:  (103, 143)
 Private key:  (127, 143)
 Enter a message to encrypt: hello
 Encrypting message using public key (103, 143) . . .
 Encrypted message is:  26140696945
 Decrypting message using private key (127, 143)  . . .
 Decrypted message is:  hello

Process finished with exit code 0
```