IZMIR INSTITUTE OF TECHNOLOGY

CENG418 – Information Security

Assignment **#1**

Deadline: 01.04.2022, 11:55 pm

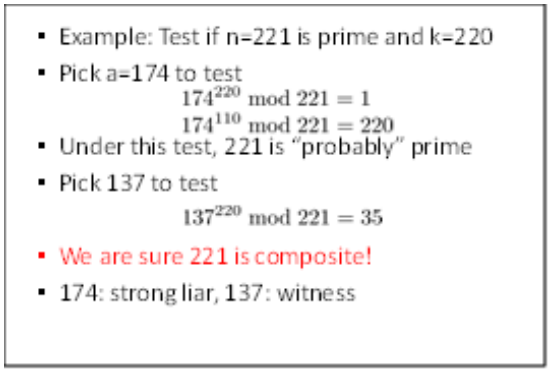
Diffie-Hellman Key Exchange (DHKE) and Secure Communication With Cipher Block Chaining (CBC)

Task Description:

1) Alice and Bob want to have a secure communication channel. To obtain a public key between the parties, they need to share a key with the DHKE method. There are some constraints for the DHKE algorithm. Alice and Bob publicly need to agree on the **prime p** and **generator g**. **As a first step of DHKE; as publicly, p and g should be shared between Alice and Bob**.

For the primality test, you can use the below algorithm:

1. Pick a random number *a,* and set *k = p - 1*
2. Calculate a^k mod *p*
3. If the result is not 1 (and if k < (n-1), result is not -1) *p* is a composite, done.
4. If the result is -1 "probably" prime, done.
5. If the result is 1 and *k* is odd, "probably" prime, done.
6. *k = k/2* go back to step 2.

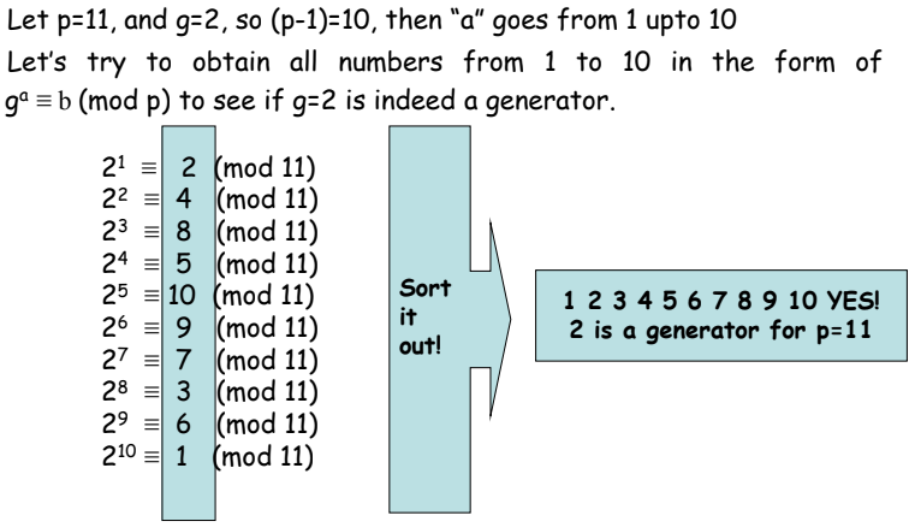


OR, you can implement your algorithm that checks whether a given number is prime or not.

Selecting generator g:

Let ***p*** be a prime with an integer ***g*** is a generator and ***g < p***. If for each integer ***b*** **from 1 to (*p*-1)**, there exists some integer ***a*** where $g^a = b \pmod p$.

Example 1:



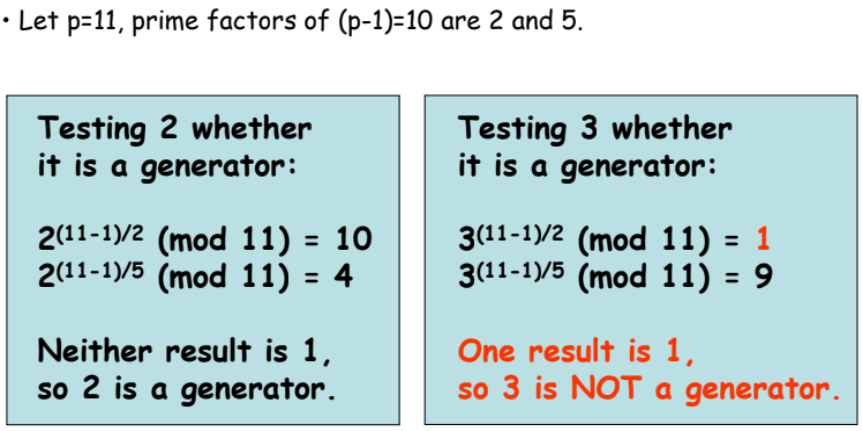
To test whether the selected g is a generator or not, you can use the algorithm below:

- Let $q_1, q_2, \ldots, q_n$ be the prime factors of *(p-1)*

  1 - Find $g^{(p-1)/q} \pmod p$ for all values of $q = q_1, q_2, \ldots, q_n$

  2 - *g* is a generator if values do not equal 1 for any values of *q*. Otherwise, it is not.

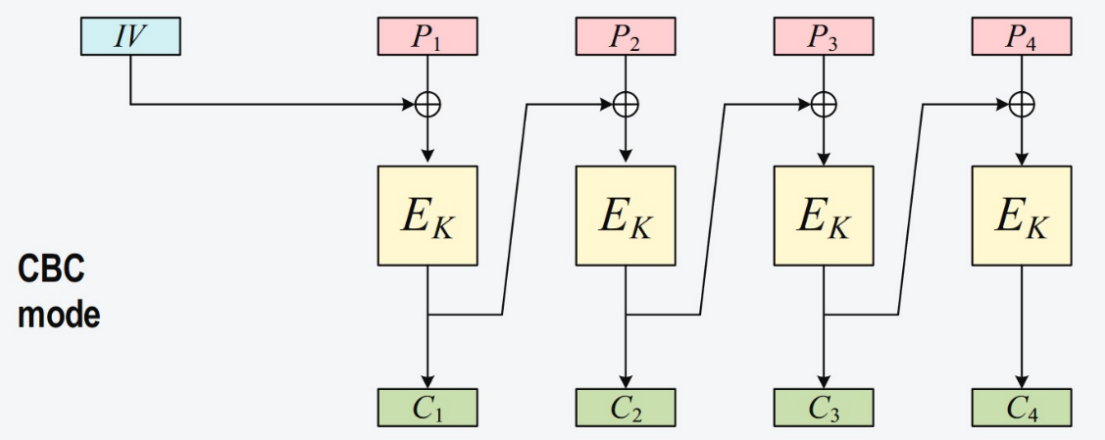
Example 2:



After determining p and g, they can generate a public key with the method with their randomly selected private keys described in the lecture notes of week 4. For modular exponentiation, you had better use **the repeated squaring** method. If you need to find the gcd (greatest common divisor) or multiplicative inverse of any number, you had better use the extended Euclidean algorithm.

By ECDH key exchange algorithm, Allice and Bob share a common secret key for their secure communication.

2) Later for secure communication, Alice encrypts her messages with the shared (common) key and sends them to Bob, and Bob will use the shared key to decrypt these messages. For both encryption and decryption, they will use the CBC illustrated below. IV is the initialisation vector that needs to be taken from the user at the beginning of the communication. IV value is also common and public value between Allice and Bob. In each block of E_K, you will use the XOR operator between common key and message to encrypt messages. Encryption and decryption will be byte byte for each character of the message. While each P_i represents the i'th message byte, each C_i represents the i'th ciphertext byte.



Note:

- **The message** will be taken from the user.
- **Private keys** for Alice **"a"** and for Bob **"b"** variables, prime ***p*** and generator ***g*** used for the DHKE **must be taken from the user**. Primality testing and generator testing should be done and approved to the user.
- One needs to verify that both sender (Alice) and receiver (Bob) use **the common key to encrypt and decrypt the message byte by byte**. Please **design UI** to show all these steps, these are important to evaluate your homework.

- Prepare a report about your modularised implementation, including your results and interpretations about the security of the design. Please add your screen snapshots to explain the details.
- Please implement your design solution in python programming language.
- Please upload your assignments in .zip format by arranging **NameSurname_ass1_CENG418.zip**