Experiment no. 3

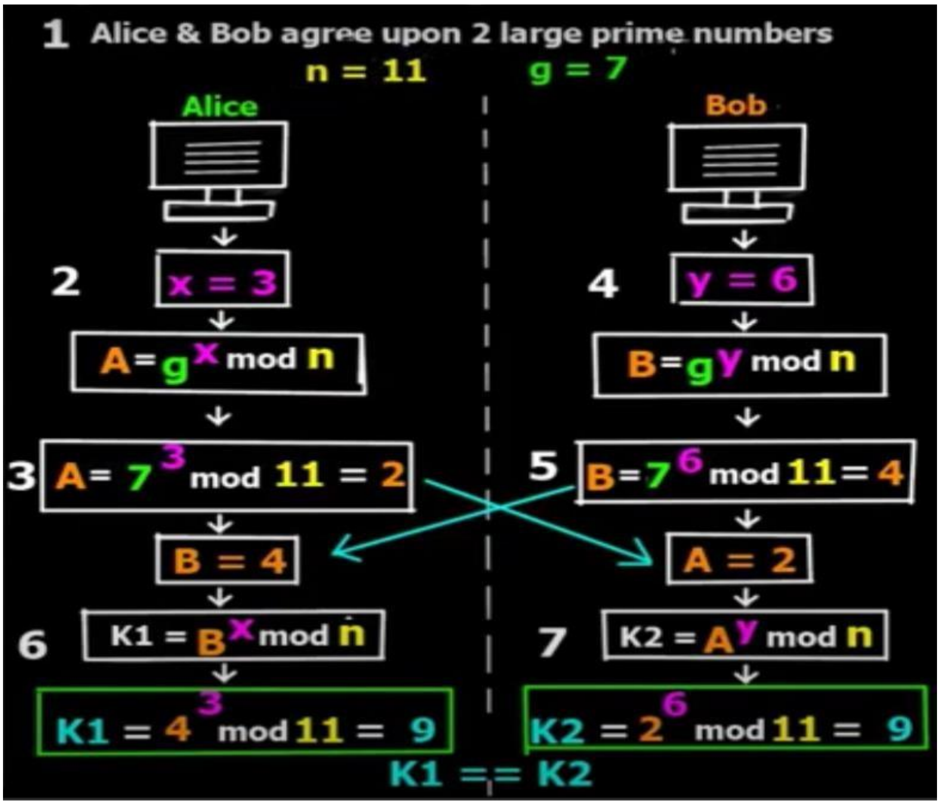
Learning Objective: Analyze and implement Diffie-Hellman Key Exchange Algorithm

Tools: C/C++/Java/Python

Theory: DIFFIE–HELLMAN KEY EXCHANGE:

Diffie–Hellman key exchange (D–H) is a specific method of exchanging keys. It is one of the earliest practical examples of key exchange implemented within the field of cryptography. The Diffie–Hellman key exchange method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure communications channel. This key can then be used to encrypt subsequent communications using a symmetric key cipher.

The Diffie–Hellman key agreement was invented in 1976 during a collaboration between Whitfield Diffie and Martin Hellman and was the first practical method for establishing a shared secret over an unprotected communication channel.



Diffie–Hellman establishes a shared secret that can be used for secret communications by exchanging data over a public network.

STEP 1: GLOBAL PUBLIC ELEMENTS:

Firstly, Alice and Bob agree on two large prime numbers, n and g. These two integers need not be kept secret. Alice and Bob can use an insecure channel to agree on them.

STEP 2: ASYMMETRIC KEY GENERATION BY USER 'A':

Alice chooses another large random number X, and calculates, the public key, A, such that:

$$A= g^{X} \bmod n$$

STEP 3: Alice sends the number A to Bob.

STEP 4: KEY GENERATION BY USER 'B':

Bob independently chooses another large random number Y, and calculates, the public key, B, such that: $B= g^{Y} \bmod n$

STEP 5: Bob sends the number B to Alice.

STEP 6: SYMMETRIC KEY (K) GENERATION BY USER 'A':

A now computes the secret key, K1 as follows: $K1= B^{X} \bmod n$

STEP 7: SYMMETRIC KEY (K) GENERATION BY USER 'B':

B now computes the secret key, K2 as follows: $K2= A^{Y} \bmod n$

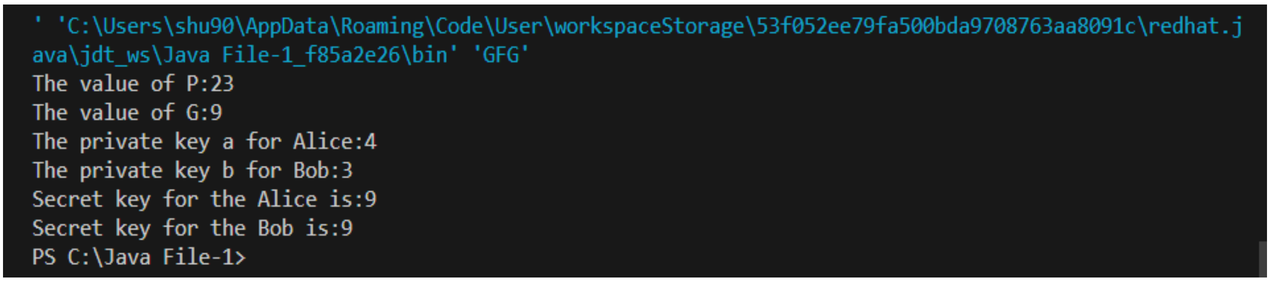
NOTE:

It should be difficult for Alice to solve for Bob's private key or for Bob to solve for Alice's private key. If it is not difficult for Alice to solve for Bob's private key (or vice versa), Eve may simply substitute her own private / public key pair, plug Bob's public key into her private key, produce a fake shared secret key, and solve for Bob's private key (and use that to solve for the shared secret key. Eve may attempt to choose a public / private key pair that will make it easy for her to solve for Bob's private key).

CODE:

```
class GFG {

// Power function to return value of a ^ b
mod P private static long power(long a, long
b, long p)
{
if (b == 1)
    return
a;else
return (((long)Math.pow(a, b)) % p);
}

public static void main(String[] args)
{
long P, G, x, a, y, b, ka, kb;



// A prime number P is
takenP = 23;
System.out.println("The value of P:" + P);

// A primitive root for P, G is
takenG = 9;
System.out.println("The value of G:" + G);

// Alice will choose the private key a
// a is the chosen private
keya = 4;
System.out.println("The private key a for Alice:"
+ a);

// Gets the generated
keyx = power(G, a, P);

// Bob will choose the private key b
// b is the chosen private
keyb = 3;
System.out.println("The private key b for Bob:"
+ b);

// Gets the generated
keyy = power(G, b, P);

// Generating the secret key after the exchange
// of keys
ka = power(y, a, P); // Secret key for
Alicekb = power(x, b, P); // Secret key
for Bob

System.out.println("Secret key for the Alice is:"
                + ka);
System.out.println("Secret key for the Bob
is:"
+ kb);
}
}
```

Output:



```
' 'C:\Users\shu90\AppData\Roaming\Code\User\workspaceStorage\53f052ee79fa500bda9708763aa8091c\redhat.j
ava\jdt_ws\Java File-1_f85a2e26\bin' 'GFG'
The value of P:23
The value of G:9
The private key a for Alice:4
The private key b for Bob:3
Secret key for the Alice is:9
Secret key for the Bob is:9
PS C:\Java File-1>
```

Result and Discussion: The Diffie-Hellman algorithm is being used to establish a shared secret that can be used for secret communications while exchanging data over a public network using the elliptic curve to generate points and get the secret key using the parameters.

Learning Outcomes: The student will be able to:

LO1: Understand the Diffie-Hellman Key Exchange Algorithm

LO2: Analyze and implement the Diffie-Hellman Key Exchange Algorithm

Course Outcomes: Upon completion of the course students will be able to analyze and implement Diffie-Hellman Key Exchange Algorithm for generation of shared symmetric key

Conclusion: The practical implementation of the algorithm, we will consider only 4 variables, one prime P and G (a primitive root of P) and two private values a and b. P and G are both publicly available numbers. Users (say Alice and Bob) pick private values a and b and they generate a key and exchange it publicly. The opposite person receives the key and that generates a secret key, after which theyhave the same secret key to encrypt.

Name: Aryan Tiwari
Class: SE – CSE
Roll.No: 57

For Faculty Use

Correction Parameters	**Formative Assessment [40%]**	**Timely completion of Practical [40%]**	**Attendance / Learning Attitude [20%]**	
Marks Obtained				