**Deffie Helman key exchange algorithm**

import java.math.BigInteger;

import java.util.Scanner;

public class DiffieHellman {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

// Accepting values of P and G

System.out.print("Enter the value of P (a prime number): ");

BigInteger P = sc.nextBigInteger();

System.out.print("Enter the value of G (a primitive root modulo P): ");

BigInteger G = sc.nextBigInteger();

// Accepting private keys a and b

System.out.print("Enter the private key a: ");

BigInteger a = sc.nextBigInteger();

System.out.print("Enter the private key b: ");

BigInteger b = sc.nextBigInteger();

// Calculating public keys

BigInteger A = G.modPow(a, P); // A = G^a mod P

BigInteger B = G.modPow(b, P); // B = G^b mod P

// Calculating shared secret keys

BigInteger sharedKeyA = B.modPow(a, P); // Shared key from A's perspective

BigInteger sharedKeyB = A.modPow(b, P); // Shared key from B's perspective

// Displaying the results

System.out.println("Public key A (G^a mod P): " + A);

System.out.println("Public key B (G^b mod P): " + B);

System.out.println("Shared key calculated by A: " + sharedKeyA);

System.out.println("Shared key calculated by B: " + sharedKeyB);

sc.close();

}

}

**Explanation:**

1. **P**: A large prime number.
2. **G**: A primitive root modulo P.
3. **a**: Private key of user A.
4. **b**: Private key of user B.
5. **A**: Public key of user A, calculated as Ga mod  P
6. **B**: Public key of user B, calculated as Gb mod  P
7. **Shared Key**: Both users calculate the shared key using each other's public key:
   * User A Computes Ba mod P
   * User B Computes Ab mod P

Both users will arrive at the same shared key, which can be used for secure communication.

Step 1: Alice and Bob get public numbers P = 23, G = 9

Step 2: Alice selected a private key a = 4 and

Bob selected a private key b = 3

Step 3: Alice and Bob compute public values

Alice: x =(9^4 mod 23) = (6561 mod 23) = 6

Bob: y = (9^3 mod 23) = (729 mod 23) = 16

Step 4: Alice and Bob exchange public numbers

Step 5: Alice receives public key y =16 and

Bob receives public key x = 6

Step 6: Alice and Bob compute symmetric keys

Alice: ka = y^a mod p = 65536 mod 23 = 9

Bob: kb = x^b mod p = 216 mod 23 = 9

Step 7:Ka=Kb=k=9 is the shared secret key.