

```
public class diffiecomplete{

    public static void main(String args[])
    {
        int p=941;
        boolean isprime = true;
        for(int i =2 ; i < p ; i ++ )
        {
            if(p%i==0){isprime = false;}

        }

        if(isprime)
        {
            System.out.println("p=" + p + " is prime");
        }
        // we have a prime apply the diffie hellman algorithm
        // refer to slide 9 and generate A' and B';
        // they should be the same number.
        int g=237, a =347;
        // What is  A = g^a mod p

        // the code below will not work xx
        // is too big
        double xx = Math.pow(627,347);
        System.out.println("xx=" + xx + "\n");

        // this code will work because it
        // removes as many multiples of p
        // as it can at each stage
        // A is
        int A=1;
        for(int k =1 ; k <=a;k++)
        {
            A = (A*g) %p;
        }
        System.out.println("A=" + A + "\n");
        // b is Bobs private key
        int B=1, b = 781;
        for(int k =1 ; k <=b;k++)
        {
            B = (B*g) %p;
        }
        System.out.println("B=" + B + "\n");
        // B is Bob's public key
        // We need to work out A' and B' (they should be
        // the same number)
        //
        int Aprime=1;
        for(int k = 1;k <=b;k++) // evaluate A^b
        {
            Aprime = (Aprime *A)%p;
        }
        System.out.println("Aprime=" + Aprime + "\n");
        int Bprime=1;
        for(int k = 1;k <=a;k++) // evaluate B^a
        {
            Bprime = (Bprime *B)%p;
        }
        System.out.println("Bprime=" + Bprime + "\n");

    }
}
```

}