

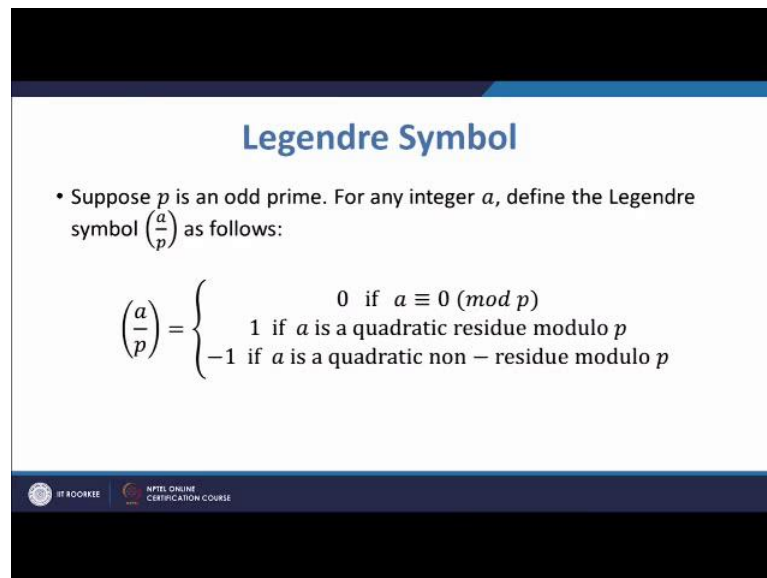
Introduction to Cryptology
Dr. Sugata Gangopadhyay
Department of Computer Science and Engineering
Indian Institute of Technology, Roorkee

Lecture - 15

Problem discussion on Jacobi symbol calculation and RSA cryptosystem

Welcome to Week 3, Lecture - 5 of our course Introduction to Cryptology. In this lecture we will review the concepts that we have already studied and solve some examples. So, the first concept that we review is Legendre Symbol.

(Refer Slide Time: 00:45)



Legendre Symbol

- Suppose p is an odd prime. For any integer a , define the Legendre symbol $\left(\frac{a}{p}\right)$ as follows:

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{if } a \equiv 0 \pmod{p} \\ 1 & \text{if } a \text{ is a quadratic residue modulo } p \\ -1 & \text{if } a \text{ is a quadratic non-residue modulo } p \end{cases}$$

IT ROORKEE NPTEL ONLINE CERTIFICATION COURSE

So, we see that if p is an odd prime and if a is any integer then the Legendre symbol, a by p is defined as follows, a by p equal to 0, if a is congruent to 0 mod p a by p is 1. If a is a quadratic residue mod p and a by p is minus 1, if a is a quadratic non residue mod p and Jacobi symbol is defined in this way, if n has factorization product p_i raise to the power e_i i running from 1 to k .

(Refer Slide Time: 01:37)

Jacobi Symbol

- Suppose that n is an odd positive integer, and the prime power factorization of n is

$$n = \prod_{i=1}^k p_i^{e_i}.$$

Let a be an integer. The Jacobi symbol $\left(\frac{a}{n}\right)$ is defined to be

$$\left(\frac{a}{n}\right) = \prod_{i=1}^k \left(\frac{a}{p_i}\right)^{e_i}.$$

IT ROORKEE NPTEL ONLINE CERTIFICATION COURSE

Then the Jacobi symbol of a mod n is defined as a product of the individual Legendre symbols raised to respective powers.

(Refer Slide Time: 01:53)

Solovay-Strassen Algorithm

SOLOVAY-STRASSEN(n)

Choose a random integer a such that $1 \leq a \leq n - 1$

$$x \leftarrow \left(\frac{a}{n}\right)$$

if $x = 0$

then return ("n is composite")

$$y \leftarrow a^{\frac{n-1}{2}} \pmod{n}$$

if $x \equiv y \pmod{n}$

then return ("n is prime")

else return ("n is composite")

IT ROORKEE NPTEL ONLINE CERTIFICATION COURSE

Now, let us look at Solovay-Strassen algorithm. Now, in this algorithm our goal is to decide, whether n is a prime number or not. So, the input is an odd prime and the

algorithm goes in this way that we choose a random integer a between an n minus 1 and then calculate the Jacobi symbol for it. So, this is Jacobi symbol a by n , we store it in x if the Jacobi symbol turns out to be 0 then we say that the number is composite.



If it is not 0 then we calculate a raised to the power n minus 1 by 2 mod n and store it in y then we check whether x and y are congruent modulo n or not if congruent mod n we say that the number n is prime otherwise not. Now, let us look at this example, now suppose that Solovay-Strassen algorithm is executed with input 561 and the random integer chosen is equal to 2 then what is the output?

(Refer Slide Time: 03:02)

Example 1

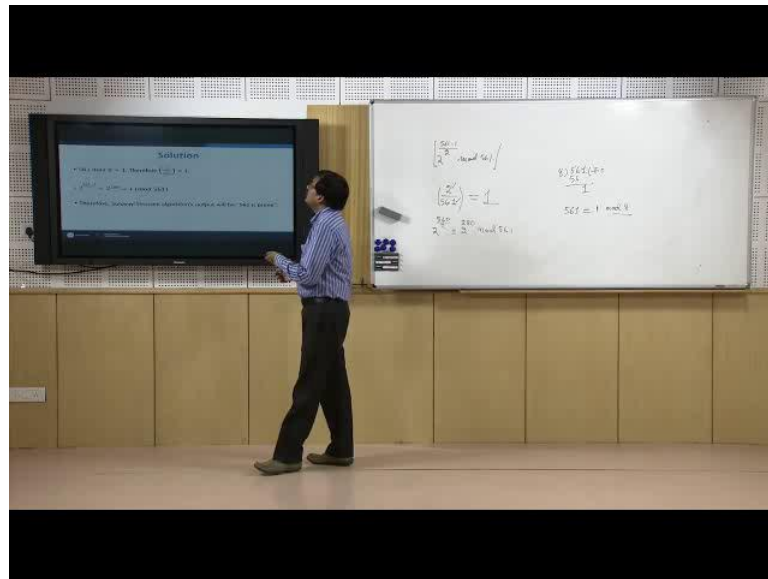
- Compute $\left(\frac{2}{561}\right) \bmod 561$.
- Compute $2^{\frac{561-1}{2}} \bmod 561$.
- Suppose that Solovay-Strassen algorithm is executed with input 561, and the random integer chosen is $a = 2$. Then what is the output?

- If n is a positive odd integer and $m_1 \equiv m_2 \pmod{n}$, then
$$\left(\frac{m_1}{n}\right) \equiv \left(\frac{m_2}{n}\right).$$
- If n is a positive odd integer, then
$$\left(\frac{2}{n}\right) = \begin{cases} 1 & \text{if } n \equiv \pm 1 \pmod{8} \\ -1 & \text{if } n \equiv \pm 3 \pmod{8} \end{cases}$$
- If n is a positive odd integer, then
$$\left(\frac{m_1 m_2}{n}\right) = \left(\frac{m_1}{n}\right) \left(\frac{m_2}{n}\right).$$
 In particular, if $m = 2^k t$ and t is odd, then
$$\left(\frac{m}{n}\right) = \left(\frac{2}{n}\right)^k \left(\frac{t}{n}\right).$$
- Suppose m and n are positive odd integers. Then
$$\left(\frac{m}{n}\right) = \begin{cases} -\left(\frac{n}{m}\right) & \text{if } m \equiv n \equiv 3 \pmod{4} \\ \left(\frac{n}{m}\right) & \text{otherwise.} \end{cases}$$

 IIT ROORKEE
  NPTEL ONLINE CERTIFICATION COURSE

So, to calculate the output we have to calculate 2 things; the Jacobi symbol 2 by 561 mod 561 and also calculate there is a miss print here we have to calculate this n is 561.

(Refer Slide Time: 03:48)



So, a is 2 here. So, have to calculate 561 minus 1 divided by 2 mod 561 is a slight error over here. So, let us neglect that now calculating the Jacobi symbol 2 by 561 . Now, at this point we see that to calculate Jacobi symbol, we do not have to calculate the factorization of 561 . We can simply neglect that and we have to only calculate, we can use the properties of the Jacobi symbol. So, we check that it is a second property. So, here the numerator is 2 , denominator is 561 .

So, in the second property we see that 2 by n is 1 if n is plus minus 1 mod 8 and 2 by n is minus 1 , if n is plus minus 1 plus minus 3 mod n . So, we have to basically check whether 561 is plus minus 1 mod 8 or plus minus 3 mod 8 . If we divide 561 by 8 then we see that this is 1 therefore, 561 is congruent to 1 mod 8 and therefore, this value is 1 . On the other hand, we have to calculate, let us look at the algorithm a raise to the power n minus 1 by 2 , here n is 561 .

So, neglecting the miss print over here, we have to calculate 2 raise to the power 561 minus 1 by 2 ; that means, we have to calculate 2 raise to the power 561 by 2 that is equal to 2 raise to the power 280 and reduce it modulo 561 we can do that by using a calculator and we will see that it comes to be 1 .

(Refer Slide Time: 07:12)

Solution

- $561 \bmod 8 = 1$. Therefore $\left(\frac{2}{561}\right) = 1$.
- $2^{\frac{561-1}{2}} = 2^{280} \equiv 1 \pmod{561}$
- Therefore, Solovay-Strassen algorithm's output will be "561 is prime".

IT ROORKEE NPTEL ONLINE CERTIFICATION COURSE

Thus, we see that 2 by 561 is equal to 2 to the power 561 minus 1 divided by 2 modulo 561.

(Refer Slide Time: 07:32)

Euler Pseudoprime

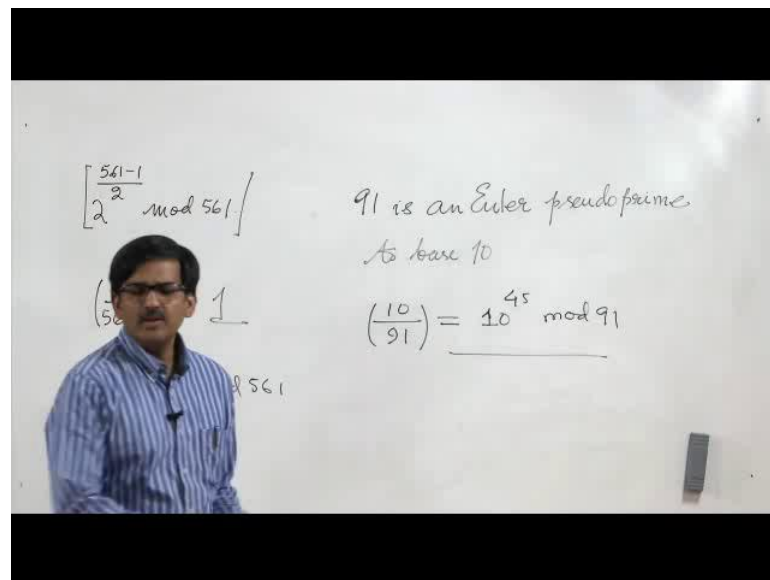
- 561 is said to be an Euler pseudoprime to the base 2.
- Similarly 91 is said to be an Euler pseudoprime to the base 2.

IT ROORKEE NPTEL ONLINE CERTIFICATION COURSE

That means, that this pair 2 and 561 satisfies this condition and that is why the Solovay-Strassen algorithm will return 561 is a prime. Although, it is absolutely clear that 561 is

not a prime, when we have such a situation then there is a name for the positive integers which are decided as primes based on certain choice of a by using Solovay-Strassen algorithm. These integers are called Euler Pseudoprime to the corresponding ways. So, in the case of 561 and the choice 2, we will say that 561 is an Euler Pseudoprime to the base 2, we have seen during our lectures that there is another pair 91. Here again 91 is not Euler Pseudoprime to base 10, it should be written as 10. So, similarly, we have seen during our lectures that ninety 1 is Euler Pseudoprime to base 10 because you can check that.

(Refer Slide Time: 09:44)



10 by 91 is equal to 10 to the power 45 modulo 91; this is something that we have done. Now, let us look at another problem, this is the problem related to computation of Jacobi symbols.

(Refer Slide Time: 10:32)


Example 2

Evaluate the Jacobi symbol:

$$\left(\frac{610}{987}\right).$$

$$\begin{aligned} \left(\frac{610}{987}\right) &= -\left(\frac{305}{987}\right) = -\left(\frac{72}{305}\right) \\ &= -\left(\frac{9}{305}\right) = -\left(\frac{8}{9}\right) = -\left(\frac{1}{9}\right) \\ &= -1. \end{aligned}$$

- If n is a positive odd integer and $m_1 \equiv m_2 \pmod{n}$, then $\left(\frac{m_1}{n}\right) \equiv \left(\frac{m_2}{n}\right)$.
- If n is a positive odd integer, then $\left(\frac{2}{n}\right) = \begin{cases} 1 & \text{if } n \equiv \pm 1 \pmod{8} \\ -1 & \text{if } n \equiv \pm 3 \pmod{8} \end{cases}$.
- If n is a positive odd integer, then $\left(\frac{m_1 m_2}{n}\right) = \left(\frac{m_1}{n}\right) \left(\frac{m_2}{n}\right)$.
In particular, if $m = 2^t r$ and r is odd, then $\left(\frac{m}{n}\right) = \left(\frac{2}{n}\right)^t \left(\frac{r}{n}\right)$.
- Suppose m and n are positive odd integers. Then $\left(\frac{m}{n}\right) = \begin{cases} -\left(\frac{n}{m}\right) & \text{if } m \equiv n \equiv 3 \pmod{4} \\ \left(\frac{n}{m}\right) & \text{otherwise.} \end{cases}$



Suppose we have got 2 numbers 610 and 987 and we would like to know the value of the Jacobi symbol 610 by 987. So, we will take this strategy, we will use the rules that we have already studied. So, let us see how to use them.

(Refer Slide Time: 11:04)

$$\begin{aligned} \left(\frac{610}{987}\right) &= \left(\frac{2 \cdot 305}{987}\right) = \left(\frac{2}{987}\right) \left(\frac{305}{987}\right) \\ &= -\left(\frac{305}{987}\right) = -\left(\frac{987}{305}\right) = -\left(\frac{72}{305}\right) \\ &= -\left(\frac{2^3 \cdot 9}{305}\right) = -\left(\frac{2}{305}\right)^3 \left(\frac{9}{305}\right) \\ &= -\left(\frac{9}{305}\right) = -\left(\frac{305}{9}\right) \\ &= -1. \end{aligned}$$

$$\begin{array}{r} 305 \overline{) 987} \\ \underline{915} \\ 72 \end{array}$$

$987 \bmod 305 = 72$

$$\begin{array}{r} 8 \overline{) 305} \\ \underline{304} \\ 1 \end{array}$$

$305 \equiv 1 \pmod{8}$

We start from miss point 610 and 987, we know that 610 is even and 305 into 2 is 610. Therefore, we note that we will split in this way, 2 into 305 and the denominator 987. So, using the third law, we will have 2 by 987 and 305 by 987. Now, ask the question that what is the value of $n \bmod 8$ that is $987 \bmod 8$? Let us find if 987 divided by 8, it is 3. So, we see that 987 is congruent to $3 \bmod 8$. Therefore, we go to the second line of the second law. So, 2 by 987 is going to be a minus 1. So, we have 305, 987. Now, we have found that $987 \bmod 8 = 3$.

So, let us see what to do with this? Now, we see that 305 is odd and 987 is also odd. So, we can directly use this reciprocity law, but we have to check which of these rules are satisfied by 987 and 305. So, we divide 305 by 4. So, we have 28 and 2. So, we see that we get 1. So, therefore, 305 is congruent to $1 \bmod 4$. So, 305 is not congruent to $3 \bmod 4$. So, therefore, we come to this case and therefore, I know that I can switch over without change of sign and that is what we have doing. Now, I will write 987, 305 and now we will use the first law to reduce 987 modulo 305, let us do that. So, it is 72.

So, we will write 72 by 305 and now let us consider 72, if we see that 72 is an even number and therefore, we have we can divide by 2. So, we know that 8 into 9 is 72. Therefore, we have 2 cubes into 9 by 305 and then using the third rule, I can write minus 2 by 305 whole cube nine by 305 all right. Now, I ask a question, what is a see value of this, for that again we have to divide 305 by 8 is 1. So, we see that 305 is congruent to $1 \bmod 8$. So, we come over here, we know that its 1. So, we have minus 1, this is 19 by 305. Now, we know that 305 is not congruent to $3 \bmod 4$ then we can again flip by using the third law. So, we will get minus 1, 305 by 9 at is what we get there and now let us reduce 305 modulo 9.

(Refer Slide Time: 18:21)

$$\begin{aligned} \left(\frac{610}{987}\right) &= \left(\frac{2 \times 305}{987}\right) = \left(\frac{2}{987}\right) \left(\frac{305}{987}\right) \\ &= -\left(\frac{2}{9}\right) \left(\frac{987}{305}\right) = -\left(\frac{72}{305}\right) \quad \begin{array}{l} 9 \mid 305 \quad 33 \\ \underline{27} \\ 35 \\ \underline{27} \\ 8 \end{array} \\ &= -\left(\frac{2^3}{805}\right) \left(\frac{9}{305}\right) \\ &= -\left(\frac{8}{9}\right) = -\left(\frac{2}{9}\right)^3 \left(\frac{1}{9}\right) = -\left(\frac{1}{9}\right) = -1 \end{aligned}$$

So, 8, I will get minus 1 minus 1 minus 8 by 9 here and this is minus of minus 2 by 9 whole cube 1 by 9 and of course, 9 is 1 mod 8. So, this is 1, I will have minus 1 by 9 and 1 by 9 is 1. So, I have minus 1 so that is the answer. So, this problem requires computation, but in some this is amazing that without factorizing, we can get information related to in somehow some information which whose definition of some quantities whose definition inverse factorization.

So, this is interesting, now a last problem, this is an algorithm that we have done in 1 of our previous lectures and this calls square and multiply algorithm and this algorithm is a fundamental algorithm that is required for practical effectiveness of RSA because in the RSA, we need exponentiation of large numbers by large numbers by, but the problem is that if we cannot do the exponentiation efficiently then we cannot work with RSA, this algorithm ensures that we can do the required exponentiations efficiently. Now, let us see how to do that.

(Refer Slide Time: 20:51)

Square and Multiply Algorithm

- $x = 7,$
- $b = 13$
 $= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2 + 1$
- $b_3 = 1, b_2 = 1, b_1 = 0, b_0 = 1.$
- $7^{13} = 96889010407$

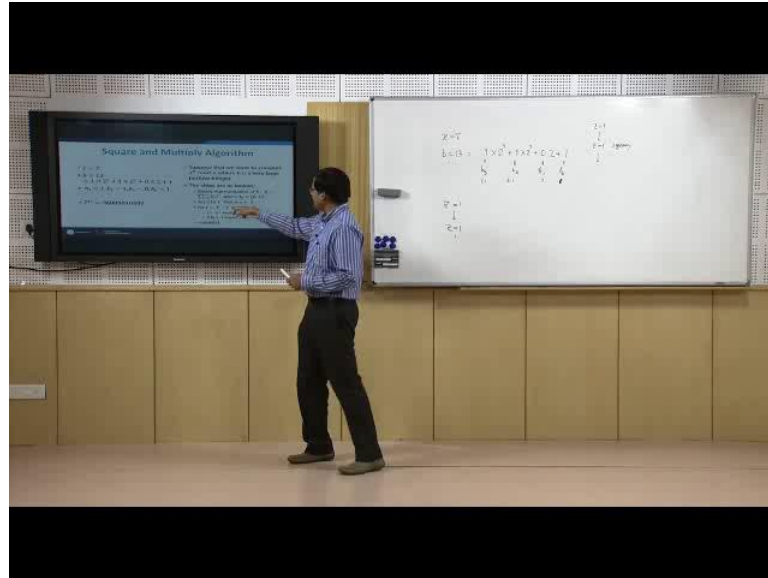
- Suppose that we want to compute $x^b \bmod n$ where b is a very large positive integer.
- The steps are as follows:
 - Binary representation of b : $b = \sum_{i=0}^{\ell-1} b_i 2^i$ where $b_i \in \{0, 1\}.$
 - Set z to 1. That is $z \leftarrow 1$
 - for $i \leftarrow \ell - 1$ down to 0
 - $z \leftarrow z^2 \bmod n$
 - if $b_i = 1$ then $z \leftarrow (z \times x) \bmod n$
 - return(z)

IT ROORKEE | NPTEL ONLINE CERTIFICATION COURSE

Now, let us recall the algorithm first, we want to compute x raise to the power $b \bmod n$, where b is a very large number and so what do we do? We take b and we write the binary representation of b , we will see how to do that and then we do something that is first we said z to 1 that is 1 is pressed in z and then we go into a loop and the loop depends on the length of b and it kind of comes down step wise and at each step i square z and then i check with the corresponding coefficient in the representation of b is 1.

If it is 1 we multiply x to the current values of z and put in z and then go back into the loop. If it is not 1 we do not do anything we go back into the loop square z and come and again going to the loop. Now, what we will do in this case is that we will take b equal to 13, which is small and x equal to 7 and see how this algorithm works all right.

(Refer Slide Time: 22:25)



So, we have x equal to 7 and b equal to 13, 13 can be written as 1 times 2 cube plus 1 times 2 square plus 0 times 2 plus 1. So, this is b_3 this is b_2 this is b_1 and this is b_0 this is 1 minus 1 1 minus 2 and so on and this is b_0 1 and 0 in my (Refer Time: 23:07) algorithm. So, in the beginning z is equal to 1 I go in always remember that it is square and multiply not multiply and square. So, we first square if we square 1 it remains as 1 then I going. So, let me write it over there. So, I start with z equal to 1 I going I squaring step. So, I am basically put in after squaring z is squaring z is 1 I go into the loop what do I see after z equal squaring a basically gone into the loop.

(Refer Slide Time: 24:02)

Square and Multiply Algorithm

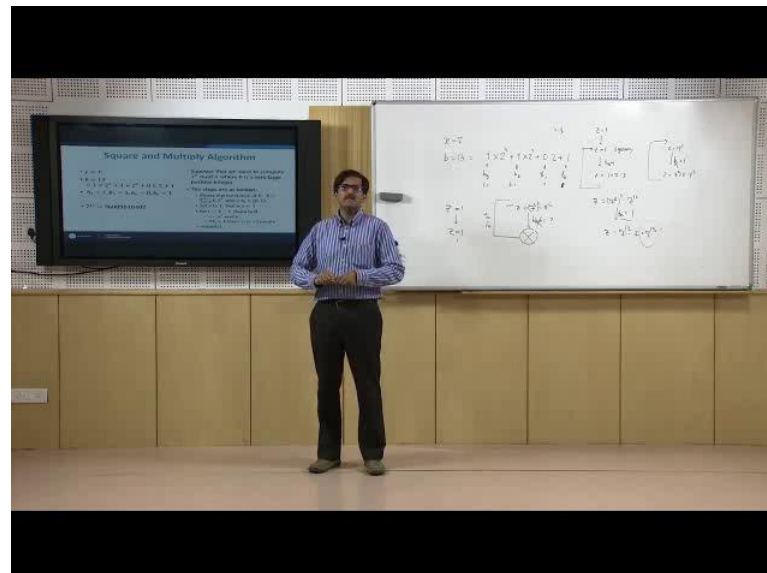
- $x = 7,$
- $b = 13$
 $= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2 + 1$
- $b_3 = 1, b_2 = 1, b_1 = 0, b_0 = 1.$
- $7^{13} = 96889010407$

- Suppose that we want to compute $x^b \text{ mod } n$ where b is a very large positive integer.
- The steps are as follows:
 - Binary representation of b : $b = \sum_{i=0}^{\ell-1} b_i 2^i$ where $b_i \in \{0, 1\}.$
 - Set z to 1. That is $z \leftarrow 1$
 - for $i \leftarrow \ell - 1$ down to 0
 - $z \leftarrow z^2 \text{ mod } n$
 - if $b_i = 1$ then $z \leftarrow (z \times x) \text{ mod } n$
 - return(z)

IT ROORKEE NPTEL ONLINE CERTIFICATION COURSE

I go into a conditional, I see whether b_i equal to 1 or not and I start from $\ell - 1$, in my case $\ell - 1$ is 3. So, when I look at it, I say that b_3 is 1.

(Refer Slide Time: 24:16)



Yes, b_3 is equal to 1, says b_3 is equal to 1; the z will be updated to 1 into 7 because x is equal to 7. So, I get 7 and then I go back again to the squaring place. Now, if I go back

again to the squaring place, my z is 7, if I square 7, what happens we get 7 square I am not writing the integer values, I am writing in terms of square. So, that we understand what happens. So, we come here, now I shift over here. So, my z is equal to 7 square, I go, I am essentially over here, I am now looping. So, once I am over here, I am asking a question has come down 1 step. So, I am asking a question whether b^2 equal to 1 or not? b^2 equal to 1 since is 1 I go into that if statement. So, z is update to 7 squares into 7.

So, that is 7 cube; I move up again to the squaring place because I am not still exhausted of I, because I has only come down to 2. So, now, I come down to 1. So, I am over here now, my z was 7 cube; I am squaring z 7 cube square is 7 raise to the power 6 and I am going down again inside the loop and I am asking a question with, whether b^1 is equal to 1 or not, b^1 is not equal to 1. So, I do not do anything, I move again back to the squaring step, what is I move back again to the squaring step, I have 7 to the plus 6 in my hand. So, here when z goes up, z is 7 to the power 6 when I come to 7 to the power 6, I square again 7 to the power 6 square is 7 to the power 12 and now I go again I ask question, whether b^0 is equal to 1 or not? b^0 is 1 over here.

So, z is updated to 7 to the power 12 into 7 which is equal to 7 to the power 13 and I am at 0. So, I go further down and therefore, my written 7 to 13. Now, in actual reality of the algorithm, what we are going to do is at each step we are going to reduce the result modulo n and we are actually going to calculate the product, but I have not done that to show that how the exponent is taking care of itself and the final value is coming to the required exponent. So, that is square and multiply algorithm.

We will have some more examples of square and multiply algorithm in the assignments and this is the end of today's lecture.