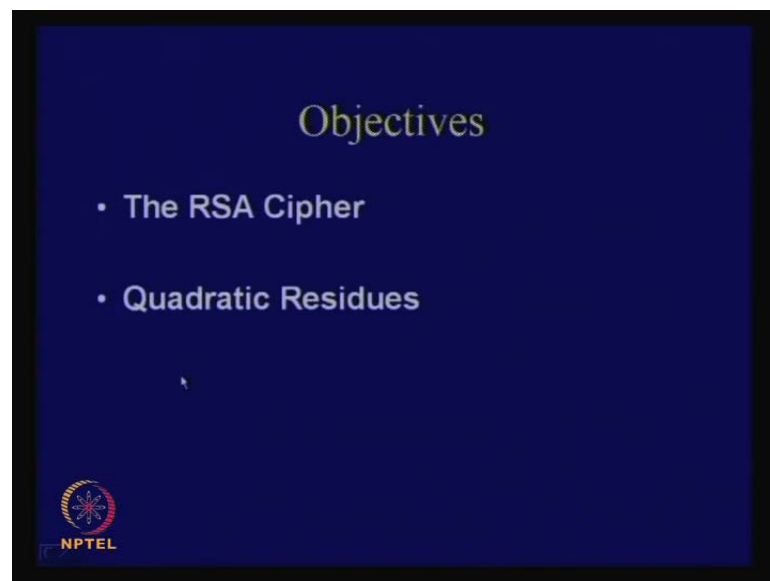


**Cryptography and Network Security**  
**Prof. D. Mukhopadhyay**  
**Department of Computer Science and Engineering**  
**Indian Institute of Technology, Kharagpur**

**Lecture No. # 28**  
**The RSA Cryptosystem**

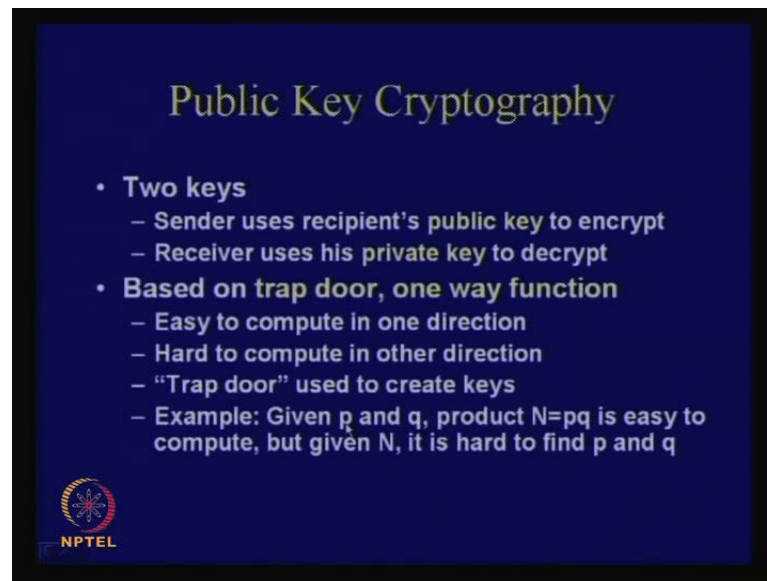
So, today, we will study the RSA cryptosystem. So, essentially we are, we of smallest discussed about the symmetric part of what symmetric cryptosystems of which is in our syllabus. So, we will now start with asymmetric cryptography.

(Refer Slide Time: 00:43)




So, with our number theoretic whatever we have learnt, we are more or less ready to understand the RSA cipher. So, today, we will take up the RSA ciphers basic definition and also to understand that whether how to operate essentially, and then, we will discuss about another interesting topic which is called quadratic residues. So, that will be also essential to understand something what is called test for primality.

(Refer Slide Time: 00:58)



**Public Key Cryptography**

- **Two keys**
  - Sender uses recipient's public key to encrypt
  - Receiver uses his private key to decrypt
- **Based on trap door, one way function**
  - Easy to compute in one direction
  - Hard to compute in other direction
  - “Trap door” used to create keys
  - Example: Given  $p$  and  $q$ , product  $N=pq$  is easy to compute, but given  $N$ , it is hard to find  $p$  and  $q$

 NPTEL

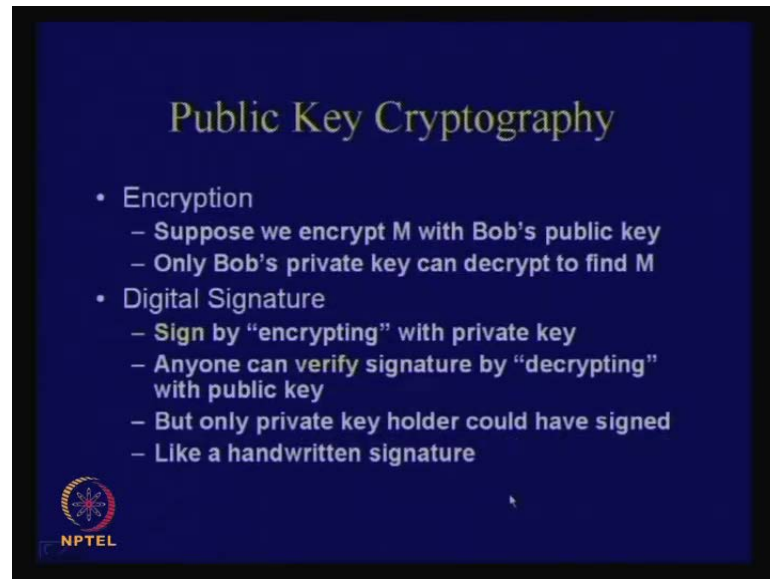
So, we will take up one-by-one. So, first of all, we know all these that, what is the definition of a public key cryptography as oppose to a symmetric key cryptosystem. So, essentially, we have got two keys. Here, we have got the sender, which uses the recipient's public key to encrypt, and the receiver, which uses his private key to decrypt. So, therefore, as oppose to the symmetric key crypto system where there was only one key share by the sender and the receiver. In this case, there are two pieces of key - one is called the public key and the other one is called the private key.

So, in order to encrypt in asymmetric cryptosystems, what we use is the public key, whereas to decrypt, we use the private key. So, this rather the security of this particular type of constructions is based upon something, which is called one way functions. So, what are one way functions? One way functions are those functions which are easy to compute but they are hard to invert, but there is something which is called a trap door one way function, because I mean for the person who does not have sudden trap door information or some secret information, for him constructing or computing the inverse should be hard.

But to a legal user who has the trap door for him, computing the inverse should be an easy problem. So, therefore, this serves as a trap door. So, an example which is commonly used is like this. Like if you take a large composite number which is made again as a product of two large prime numbers, then it is believed that factorization of  $N$

is actually a very hard problem, because therefore, if I give you  $p$  and  $q$  from their computing,  $n$  is easy, but give you  $N$  from there, finding out or ascertaining  $p$  and  $q$  is a difficult problem. But if you are given one of the factor say  $p$ , then computing the other factor is quite easy. So, that solves something like a trap door. So, therefore, this is the basic idea behind one way functions and trap door one way functions.

(Refer Slide Time: 03:16)



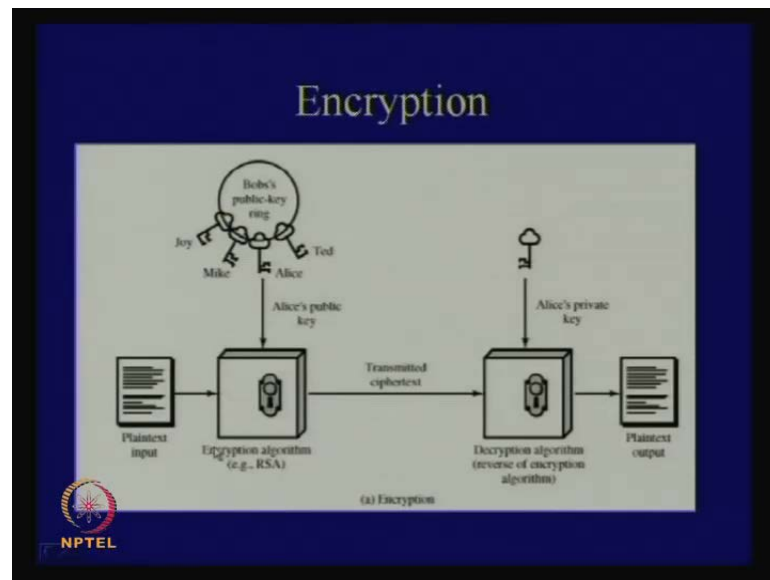
So, often public key cryptosystems are also used for something which is called digital signatures. So, in case of digital signatures as you can understand, the objective is something to do like the handwritten signature. So, therefore, for example, I have an handwritten signature and that is supposed to be like another person should not be able to copy that signature.

So, the idea is that in this case, since you have the private key, we just not supposed to be disclosed to every, I mean, I mean all people do not know the public key. So, therefore, in this case, when I will encrypt, I will encrypt using the... So, therefore, I will, in order to do the signature, I will use the private key, but in order to verify whether the private key is really my private key, you will use a public key.

So, actually the private key and public key are suppose, I mean like an inverse of each other, but you can apply in any order, like you can apply the private key first or the

public key first but it depends upon your application. So, when you are doing encryption, then actually you are using public key to encrypt.

(Refer Slide Time: 04:30)



But when you are doing signatures, there actually you are using private key to do the signals and the reason is quite easy because of the application. So, this is the pictographic view, that is, when you are trying to encrypt, then what you do is, that you take the plaintext and you have got all. Say suppose I want to send it to a person; so, therefore, suppose I want to encrypt, therefore, I have to encrypt using public.

(C)

And in order to decrypt, then I will use the private key. Therefore, always has a corresponding private key, but from this key ring which has got all the public keys like whether I want to talk to Joy or Mike or Alice or Ted or anybody else. So, therefore, what we can do is that, from whatever, I mean whom I required to contact or communicate with, I will choose the appropriate public key and then encrypt the corresponding plaintext message, and then, the cipher text is being transmitted and Alice, because Alice knows only the private key, no. It is assumed that nobody else except from for Alice can actually decrypt the information.

So, therefore, you see that there are certain things to be understood from here like how Alice generates the public key and private key pair is actually very central to this algorithm.

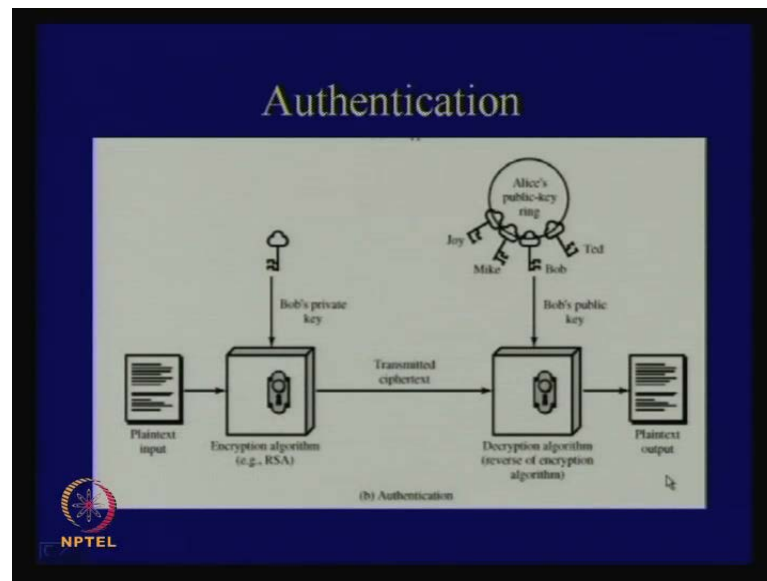
Why does Alice require a public key?

Why does Alice require a public key? So, therefore, the idea is that you know the definition of symmetric key crypto systems. So, in that case, what we are assumed is that both the sender and the receiver shares the particular piece of information, but there is a problem, that is, you have to do the key exchange, because both of, both of the, I mean sender and the receiver has to establish the same piece of key but that is not so easy because you have to essentially have some costly channel for that through which you can settle out the key.

So, actually public key encryption was actually, I mean it was around 1976 at that, that, time actually when these type of ciphers came into existence, I mean in the literature at least, because the advantage is that, here, you can actually do without a key exchange. That is a very fundamental advantage. Why? You know, because here, the idea is like this, that is, suppose Alice generates a private key which he does not communicate to Bob or at to anybody else, but what he or she publishes is actually the public key. So, from private key to compute the public key is quite easy but from public key to compute the private key is a computational difficult problem.

So, therefore, they Alice the under locking security of this type of algorithms and that is the advantage also. So, therefore, now you essentially have got this private key through which you can decrypt and it is assumed that nobody else but Alice can decrypt, because Alice only has the private key, and from this public key information, I mean any (( )) who will try to obtain the private key will actually face a quite appeal task, which should not be allowable in the today's computational context.

(Refer Slide Time: 07:43)



So, this is the basic idea, so, where public key encryption fits in. So, for authentication, when you are doing a signature, you are doing just the opposite. Therefore, here, when Bob writes I mean generates, so, Bob what he does is that, he takes his private key and he generates a cipher text, but this verification process can be done by corresponding public key.

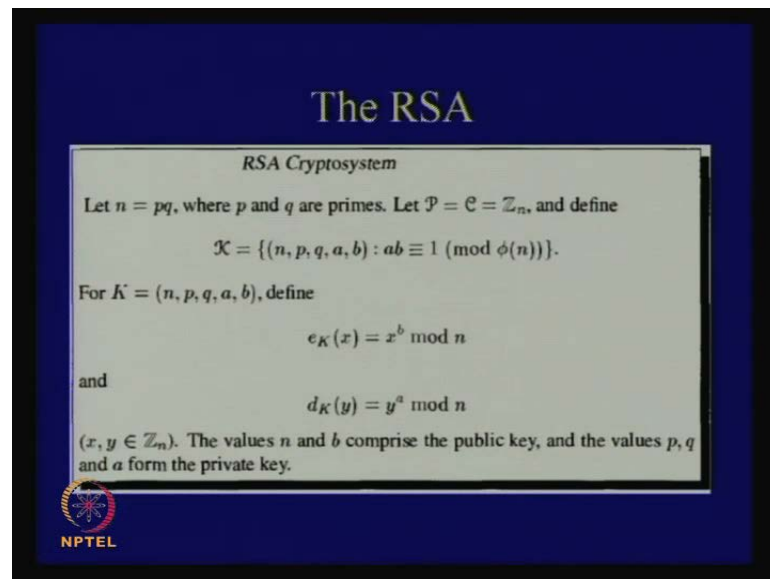
So, anybody can verify but sign only Bob can sign it because Bob only has the private key. So, that is the basic story or the basic application of public key ciphers and we will also study, I mean for example, the RSA algorithm which came into existence by the three great people like, you know, Rivest, Shamir and Adleman. So, they invented this RSA algorithm and this RSA's underlined security is based upon a problem which is called factorization, but there are actually an array of other not large, not so large also but still a quite a few, I mean for example, you have got the discrete logarithm problem which we will study also is based upon something and there is something called discrete logarithmic problem.

So, there are certain computationally assumed to be computationally hard problems based upon which, asymmetric cryptography are constructed. So, when you make an asymmetric cryptographic algorithm, even knew cryptographic algorithm, the first thing is to find out such a hard problem.

And also it is not so easy, because you not only have to find one hard problem, we just supposedly a good one way function or a candidate for a one way function, but at the same time, you also need a trap door one way function because you need that a legal person should be able to retrieve the data, should be able to decrypt the data so that makes it quite challenging.

And at the same time, I mean constructing one way functions I know, I know, that even after so much amount of work, people are not been able to really prove that it is a one way function. It is only like it is believe to be a one way function. Therefore, it is called a candidate for a one way function, but we do not really know this is the track or one way function. So, therefore, let us go into the RSA algorithm. Therefore, you have got this RSA crypto system. So, therefore, here actually you take two prime numbers  $p$  and  $q$  which are actually significantly large numbers.

(Refer Slide Time: 09:46)



The RSA

*RSA Cryptosystem*

Let  $n = pq$ , where  $p$  and  $q$  are primes. Let  $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$ , and define

$$\mathcal{K} = \{(n, p, q, a, b) : ab \equiv 1 \pmod{\phi(n)}\}.$$

For  $K = (n, p, q, a, b)$ , define

$$e_K(x) = x^b \pmod{n}$$

and

$$d_K(y) = y^a \pmod{n}$$

$(x, y \in \mathbb{Z}_n)$ . The values  $n$  and  $b$  comprise the public key, and the values  $p, q$  and  $a$  form the private key.

NPTEL

So, large means suppose around 1,024 bits. So, typically I will just give you some thumb rule data actually. That is it. So, therefore, you take two say around say 1024 bit value  $n$  and which can be also decomposed into  $p$  and  $q$  of similar kind of dimensions. So, here, your  $p$  and  $q$  are two prime numbers. If you multiply them, you get  $m$ , but there is a clean material which is actually made up of the public key part and also the private key part.

So, here, you have got  $n$  which is actually the public key information, everybody knows what is the value of  $n$ ;  $n$  is published actually but  $p$  and  $q$  are secret information's. So, suppose I want to encrypt some data, I have my own  $p$  and  $q$  values.

So, there is something which is called  $a$ , which is actually also the, I mean part of the private key and that is  $a$   $b$  which is actually the public. So, here, you have got two materials like  $a$  and  $b$  which, if you multiply and if you take mod of  $\phi n$ , then you get 1. Therefore,  $a$  and  $b$  are the multiplicative inverses of each other in the field of, what is the field? But modulo  $\phi n$ , but what is a field?

$Z_n^*$

$Z_n^*$ . Therefore, you are doing a modulo  $\phi n$  there, and therefore, for  $K$  equal to  $n$   $p$ ,  $q$ ,  $a$ ,  $b$ , there are two parts - the values  $n$  and  $b$  comprise the public key, whereas the other part that is the values  $p$   $q$  and  $a$  forms the private key. So, what is the encryption function? The encryption function looks fairly easy. You take  $x$  which you want to encrypt. So, as I told that  $b$  is the public key. Therefore, if you want to encrypt  $x$ , what you do is that you take  $x$  and raise it to the power of  $b$  and take a modulo  $m$ , and if you want to decrypt, then you take  $x$  to the power of  $b \text{ mod } n$  which is  $y$ , and then, what you do is that, you raise it to the power of  $a$  in mod  $n$ .

So, the belief is that, therefore, what you have to now prove is that, if you do decay  $y$ , you get back to  $x$ . That is what you need to prove. So, the values  $n$  and  $b$  comprise the public key and the values  $p$ ,  $q$  and  $a$  forms the private key, is the definition clear?



(Refer Slide Time: 09:46)

**The RSA**

*RSA Cryptosystem*

Let  $n = pq$ , where  $p$  and  $q$  are primes. Let  $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$ , and define

$$\mathcal{X} = \{(n, p, q, a, b) : ab \equiv 1 \pmod{\phi(n)}\}.$$


For  $K = (n, p, q, a, b)$ , define

$$e_K(x) = x^b \pmod{n}$$

and

$$d_K(y) = y^a \pmod{n}$$

$(x, y \in \mathbb{Z}_n)$ . The values  $n$  and  $b$  comprise the public key, and the values  $p, q$  and  $a$  form the private key.

  
NPTEL

So, there are certain things to be kept in mind like  $a$  and  $b$ , if you multiply and then take modulo  $\phi(n)$ , you get back one encryption function is  $x$  raised to the power of  $b$  - where  $b$  is the public key mod  $n$ , and if you are decrypting, then you raise it to the power of  $a$ . So,  $a$  has to be the private key part and you do a model. So, actually you see that whether  $b$  is the public key or  $a$  is the public key does not really matter. If it holds because since multiplication is a commutative process. Therefore, really does not matter where  $b$  is a private key or  $b$  is a public key.

But the thing is that when you are encrypting, then you use  $b$  as a public key, but when you are generating digital signatures, then  $b$  will be a private key. So, it just depends upon the application, but mathematically anything can work. So, we have to prove this result.

(( ))

Yes

(( ))

It comes. First of all, let us see the proof then, then you will understand actually. So, actually in the definition of the encryption function, it really does not matter but it matters in case of  $a$  or  $b$  equal to 1. So, what is  $\phi(n)$ ?

(C)

No, but phi n, that is fine, but what is phi n? Computationally it is equal to

Number of p minus 1

P minus 1 into q minus 1. So, it comes in the definition phi n means p minus 1 into q minus 1. So, there p and q comes. So, let us try to understand the proof. The proof can be actually divided into two parts.

(Refer Slide Time: 14:04)

**Proof of Correctness**

$ab \equiv 1 \pmod{\phi(n)} \Rightarrow ab = 1 + t\phi(n)$   
for some integer  $t \geq 1$ .

Suppose,  $x \in Z_n^* \Rightarrow x^{ab} \equiv x^{1+t\phi(n)} \equiv x(x^{\phi(n)})^t \equiv x \pmod{n}$   
[follows from Euler's Theorem]

Now, consider  $x \in Z_n \setminus Z_n^*$   
So,  $\gcd(x, n) \neq 1 \Rightarrow (x \text{ is a multiple of } p) \text{ or } (x \text{ is a multiple of } q)$   
Thus,  $\gcd(x, p) = p$  or  $\gcd(x, q) = q$   
If  $\gcd(x, p) = p$ , then  $\gcd(x, q) = 1$   
[as otherwise  $x$  is a multiple of both  $p$  and  $q$  and still  
less than  $n = pq$ ]

NPTEL

So, first, you see that a b equal to 1 or congruent 1 mod phi n means the a b of this, of this type that 1 plus t of phi n - where t is some integer greater than equal to 1. Therefore, if you take a mod of phi n, then you will get back 1. So, suppose x belongs to Z n star. Therefore, x belongs to Z n star means if I raise x to the power of a b; then that means that is x to the power of 1 plus t phi n and this you can decompose into x into x to the power of phi n whole power t, and, you know, by your previous result like last day we saw x to the power of phi n mod n is equal to 1.

So, using that result, you get that x mod n. So, this follows from Euler's theorem, but if x does not belong to Z n, I mean whether it does not belongs to Z n star; so, whether basically it belongs to Z n difference Z n star, then the proof is slightly trickier. So, let us

see this. So, this part is clear? This is called straightforward follows directly from the Euler's theorem.

So, in this part, so, if  $x$  does not belong to  $Z_n^*$ , that means the  $\gcd$  of  $x$  and  $n$  is not equal to 1 because  $x$  and  $n$  are not common sub expression-prime. If  $x$  would have belong to  $Z_n^*$ , then  $x$  would have been common sub expression-prime to  $n$ , but if  $x$  does not belong to  $Z_n^*$ , it implies that  $x$  and  $n$ , if you take the greatest common divisor, it is not equal to 1. Now, what is  $n$ ?  $n$  is the product of two prime numbers  $p$  and  $q$ . So, if, in that case, if you are and remember the  $x$  is also less than  $n$ ; so, that means what? That means that  $x$  has to be either a multiple of  $p$  or it has to be a multiple of  $q$ .

So, in this case, let us assume that the  $\gcd$  of  $x, p$  can be either  $p$  or  $\gcd$  of  $x, q$  will be equal to  $q$ . So, because if  $x$  is a multiple of  $p$ , then the  $\gcd$  of  $x, p$  will be  $p$ , and if the  $\gcd$  of  $x$  and  $q$  is equal to  $q$ , then  $\gcd$  of  $x, q$  will be equal to  $q$ . Note another thing that, if the  $\gcd$  of  $x, p$  is equal to  $p$ , then the  $\gcd$  of  $x, q$  has to be equal to 1, why, because otherwise  $x$  will be a multiple of both  $p$  and  $q$ , and therefore, it will be more than  $n$  but we know that  $x$  is by definition less than  $p, q$ , because it belongs to  $Z_n$ . It is  $x$  belongs to  $Z_n$  difference  $Z_n^*$ ; so,  $x$  belongs to  $Z_n$ .

So, therefore,  $x$ , therefore,  $\gcd$  of  $x, p$  will be equal  $p$ , then it implies that  $\gcd$  of  $x, q$  is equal to 1; that means what? That  $x$  and  $q$  are mutually common sub expression-prime. So, if  $x$  and  $q$  are mutually common sub expression-prime, then I can apply my from Euler's simple theorem, and what is that?

$x$  to the power

$x$  to the power of  $\phi(q)$

$\phi(q)$

And  $\phi(q)$  is  $q - 1$  because  $q$  is a prime number, and therefore, that should be equal to

1  $(())$

(Refer Slide Time: 17:12)


**Proof of Correctness**

Thus,  $x^{\phi(q)} \equiv 1 \pmod{q} \Rightarrow x^{t\phi(q)} \equiv 1 \pmod{q}$   
 $\Rightarrow x^{t\phi(q)\phi(p)} \equiv 1 \pmod{q}$   
 $\Rightarrow x^{t\phi(n)} \equiv 1 \pmod{q}$

Thus,  $x^{t\phi(n)} = 1 + kq$ ,  
where  $k$  is a positive integer  
Multiplying both sides by  $x$ ,  
 $x^{t\phi(n)+1} = x + kqx$

Q  $\gcd(x, p) = p \Rightarrow x = cp$ , for some positive integer  $c$   
 $x^{t\phi(n)+1} = x + kcpq$   
 $\Rightarrow x^{t\phi(n)+1} \equiv x^{ab} \equiv x \pmod{n}$

Similarly, we can prove when  $\gcd(x, q) = q$



1, and therefore, we apply this that is  $x$  to the power of  $\phi(q)$  equal to  $1 \pmod{q}$  and we get this as, therefore,  $x$  to the power of  $\phi(q)$  equal to  $1 \pmod{q}$ , I can raise it to the power of  $t$ . So,  $t$  is any integer and that still remains one. So, what about this? That  $x$  to the power of  $\phi(q)$ , I mean  $t\phi(q)$ , if I raise to the power of  $\phi(p)$ , that still remains one, and, you know, that  $\phi(p)$  into  $\phi(q)$  is what  $\phi(n)$ . So, that means  $x$  to the power of  $t\phi(n)$  is congruent to  $1 \pmod{q}$ .

So, that means I can write this  $x$  to the power of  $t\phi(n)$  as  $1 + kq$  - where  $k$  is some positive integer. So, multiplying both sides by  $x$ , if I multiply both the sides by  $x$ , then I get  $x$  to the power of  $t\phi(n) + 1$  equal to  $x + kqx$ . Now, use this is since somehow this software does this thing. So, therefore, this since  $\gcd(x, p)$  is equal to  $p$  because we are assume that  $\gcd(x, q)$  is equal to  $1$ . So,  $\gcd(x, p)$  equal to  $p$ , and therefore,  $x$  is actually a multiple of  $p$ . So, if you take this and if you substitute here, you get  $x + kcpq$ .

So, now, if you take a mod of  $pq$  on both sides, you get back only  $x$ . So, you see that  $t\phi(n) + 1$  is nothing but a  $b$ , and therefore,  $x$  to the power of  $a$  is equal to  $x$  or congruent to  $x \pmod{n}$ . Similarly you can prove the other case also. So, are we convinced about the proof? So, therefore, the RSA holds whenever  $x$  belongs to  $\mathbb{Z}_n$  in that case. So, we are prove that, it holds when  $x$  belongs to  $\mathbb{Z}_n^*$ ; we have also prove


that it hold when  $x$  belongs to  $Z_n$  difference  $Z_n^*$ . So, it belongs to whenever  $x$  belongs to  $Z_n$ .

So, what we are proved is that, if you take  $x$ , if you encrypt it and if you decrypt it by the corresponding inverse in  $Z_n^*$ , that is, when you are taking a mod of  $\phi(n)$ , then actually what you obtain is the plain, the plain text with which you started. So, what we have proved is the inevitability of the RSA transformation. That is the encryption or the decryption or inverse of one or one another.

(Refer Slide Time: 19:47)

**Example**

- Bob chooses  $p=101$  and  $q=113$ 
  - Thus  $n=11413$
  - $\Phi(n)=100 \times 112=11200=2^6 5^2 7$
  - $b$  can be used for encryption if and only if it is not a multiple of 2, 5 or 7. Let  $b=3533$
- In practice Bob will not factor  $\Phi(n)$ , but will check whether  $\gcd(b, \Phi(n))=1$  using EA and compute  $b^{-1}$  at the same time.

 NPTEL

So, this is a simple example of  $p$  equal to 101 and  $q$  equal to 103. Therefore, this is actually this is the RSA algorithm being shown but actually this is not secured, why, because  $p$  and  $q$  are too small. So, therefore, these are all insecure RSA example. So, you can say that in if I multiply it will be 11413 and  $\phi(n)$  works to 100 into 112. So, just believe that it is correct now, but you can see this because hundred one and hundred

9


112, therefore, it follow straight. So, this you can actually factor. Therefore,  $2^6 5^2 7$  square and 7. So, now,  $b$  can be used for encryption if and only if it is not a multiple of 2 5 or 7, why? Yes because  $b$  has to have an inverse, multiplicative inverse, and therefore,  $b$  has to be mutually co-prime with  $\phi(n)$ .

So, if  $b$  has to have because  $a \cdot b$  is congruent to  $1 \pmod{\phi(n)}$ . Therefore,  $b$  has to be mutually common sub expression-prime with  $\phi(n)$ . Therefore,  $b$  should not be divisible by 2, 5 or 7. So, let  $b$  be equal to 1 value say 3533. In practice, Bob will not factor  $\phi(n)$  but will check whether  $\text{gcd}(b, \phi(n)) = 1$  using the Euclidean algorithm, and, you know, that it is quite an efficient algorithm, it has got a polynomial run time.

(Refer Slide Time: 21:10)

**Examples**

- Bob publishes  $n=11413$  and  $b=3533$ .
- Suppose Alice wants to encrypt  $x=9726$  and send to Bob.
- Hence, she computes  $x^b \pmod{n}$   
 $= 9726^{3533} \pmod{11413} = 5761$  and sends it to Bob.
- Bob computes  $b^{-1} \pmod{\phi(n)} = 6597$  and decrypts using  $5761^{6597} \pmod{11413} = 9726$

 NPTEL

And we will compute the  $b$  inverse at the same time. That is the way how we will get the  $b$  inverse. So, Bob what he does is that, Bob publishes  $n$  equal to 11413 because that is a product of the two prime numbers and also the corresponding public key. So, suppose Alice now wants to send some information say  $x$  equal to 9726 to Bob. So, what she does is that, she computes  $x$  to the power  $b \pmod{n}$ . So, therefore, that is 9726 to the power of 3533 which is the public key and does a mod of  $n$ . So, mode 11413 and what she computes is 5761 and send it to Bob.

So, now, Bob needs compute  $b$  inverse mod  $\phi(n)$ . So, he engages, the Euclidean extended Euclidean algorithm do that and computes and finds out 6 5 9 7 and then decrypts it using 5761 to the power of 6597 and believe me it reduces to 9726. Therefore, you get back whatever you started with; that is a basic idea.

(Refer Slide Time: 22:11)

**Efficient Exponentiation**

- Compute  $x^c$  efficiently mod  $n$ .
- Express  $c$  as follows:  $c = \sum_{i=0}^{\ell-1} c_i 2^i$

```
SQUARE-AND-MULTIPLY( $x, c, n$ )  
 $z \leftarrow 1$   
for  $i \leftarrow \ell - 1$  downto 0  
do {  
   $z \leftarrow z^2 \bmod n$   
  if  $c_i = 1$   
  then  $z \leftarrow (z \times x) \bmod n$   
}  
return ( $z$ )
```

NPTEL

So, now, the next question which comes to our mind in terms of implementation is that, how do I compute this  $x$  to the power of  $c$  efficiently, because I need to compute the powers. So, one good algorithm which I have doing that, I mean first in order to start what we can do is that, we can multiply  $x$  say  $c$  times, but you understand that that will actually not be very efficient. So, what we can do is that we express  $c$  in the binary format, and then, we can employ an algorithm which is the square and multiply algorithm. So, it is quite a simple algorithm.

(Refer Slide Time: 22:48)

$7_{10} = 111_2$       $5_{10} = 101_2$

$x = 1$

$\left\{ \begin{array}{l} x = 1^2 = 1 \\ x = 1 \times x = x \end{array} \right.$

$\left\{ \begin{array}{l} x = x^2 \\ x = x^2 \times x = x^3 \end{array} \right.$

$\left\{ \begin{array}{l} x = x^6 \\ x = x^6 \times x = x^7 \end{array} \right.$

$x = 1$

$\left\{ \begin{array}{l} x = x \\ x = x^2 \\ x = x^4 \\ x = x^5 \end{array} \right.$

NPTEL

So, I will just, whatever I mean give a sketch of this. So, suppose you need to compute  $x$  power 7. So, what you do is that, I take 7 and I express 7 in the binary format. So, 111 is the corresponding binary format for 7. So, what you do is that, whenever you see a 1, so, each either is like this that, each time you start with  $x$  equal to 1. So, each time whether you see a one or whether you see a 0; so, first of all you parsing the digits here like this. So, whenever you see a 0 or whether you see a one, you are always doing a squaring operation.

But if you see a one, then you do a multiplication operation multiplication means multiplication with  $x$ . So, therefore, what you do is that, you take  $x$  equal to 1 and since you see here a 1, so, what you do is that you compute 1 square. So, first of all  $x$  will be equal to 1 square which is 1, but since there is a 1, therefore you will also multiply this with  $x$ . So, you get  $x$ . In a second stage, you see again a 1. So, you do  $x$  equal to 1 square equal to, I mean whether  $x$  equal to  $x$  square because there is  $x$  you square it, and then, you multiply the current  $x$  with  $x$ . So, you get  $x$  to the power of 3.

In a second case, you again see a 1. So, what you do is that you take  $x$  and you square it, so, it becomes  $x$  power 6, and then, since so 1, you multiply this with  $x$ . So, you get  $x$  power 7. You see that this will also work, I mean let us take another example say 5. 5 is equal to 101 and see how it works. So, here, again, you start with  $x$  equal to 1, and since you get a one, the first step will be again  $x$  equal to  $x$ . So, I am just combining these two steps. In the second stage, you see a 0. So, you will do only the square part and not the multiplication part. In the third, again you see 1. Therefore, you do what you do  $x$  power 4 and then you multiply this with  $x$ . So, you get  $x$  power 5.

So, in this case, you see that the maximum number of times you need to do this operation is actually the bit length of the corresponding exponent value. So, you can actually do it more efficient. So, what will be the run time of this? If  $p$  is your maximum number of values to the in the field, then it will be  $\log p$ . So, therefore, it will be  $\log p$  base 2. So, therefore, this has got this is more an efficient algorithm. There are ways how you can actually probably try to increase the efficiency but this is the basic, I mean a skeletal algorithm.



(Refer Slide Time: 25:38)

**Choosing the parameters of RSA**

**RSA PARAMETER GENERATION**

1. Generate two large primes,  $p$  and  $q$ , such that  $p \neq q$
2.  $n \leftarrow pq$  and  $\phi(n) \leftarrow (p-1)(q-1)$
3. Choose a random  $b$  ( $1 < b < \phi(n)$ ) such that  $\text{gcd}(b, \phi(n)) = 1$
4.  $a \leftarrow b^{-1} \pmod{\phi(n)}$
5. The public key is  $(n, b)$  and the private key is  $(p, q, a)$ .

- $n$  is known, but its factors are not known
- $b$  is also known, so to compute  $a$  one needs the value of  $\phi(n)$ , for which we need  $p$  and  $q$
- It has been conjectured that breaking RSA is polynomially equivalent to factoring  $n$ . But there is no proof!

Typically, value of  $n$  is 1024 bit long and the factors are also large of around 512 bits.

NPTEL

So, next thing that we need to do is that, we need to compute the parameters of RSA. So, parameters of RSA means what? You have to choose the corresponding values. Therefore, what we do is like this that you generate two large prime numbers  $p$  and  $q$  such that  $p$  is not equal to  $q$ . So, and what is  $n$ ?  $n$  is the product of  $p$  and  $q$  and  $\phi(n)$  is equal to  $(p-1)(q-1)$ . Now, you have to choose a random  $b$  such that the  $\text{gcd}$  of  $b, \phi(n)$  has to be equal to 1. Then compute the inverse of  $b \pmod{\phi(n)}$  and then the public key is you publish is  $(n, b)$  and the private key which you publish is  $(p, q, a)$ .

Sir

Is it for the  $n, b$  you can always calculate try to calculate  $a$ ?

From where

In an  $n$  and  $b$

From  $n$  and

$B$  as a public key cannot you compute that  $a$  equal to  $b$  inverse mod  $\phi(n)$ .

Yes, that you cannot because of the  $\phi(n)$  thing. So, that is the nice question. If you had not ask, then I would have asked in during the exams. So, can you think of why it will not work?

Because if I know the  $n$  phi  $n$  is already known.

Phi is known

(Refer Slide Time: 27:01)

$$\boxed{n = pq}$$
$$\phi(n) = (p-1)(q-1).$$
$$\phi(n) = pq - \underbrace{(p+q)} + 1.$$

Is phi  $n$  known, because see, what you will know is you know  $n$  equal to  $p$ , you know  $n$  equal to  $p$   $q$ . So, what is phi  $n$ ? Phi  $n$  is...

Phi  $n$  is a number of try in number numbers which are actually less than and which are co-prime to  $n$ . So, I can always find the kind of numbers.

No, that is why, that is why the size of  $n$  is 1,024. So, in that particular thing, so the, with so many values that approach would not work, and computationally also you see that phi  $n$  equal to  $p$  minus 1 into  $q$  minus 1. So, therefore, if you can find out phi  $n$ , then what does it mean? It means that you can write this phi  $n$  as  $p$   $q$  minus  $p$  plus  $q$  plus 1.

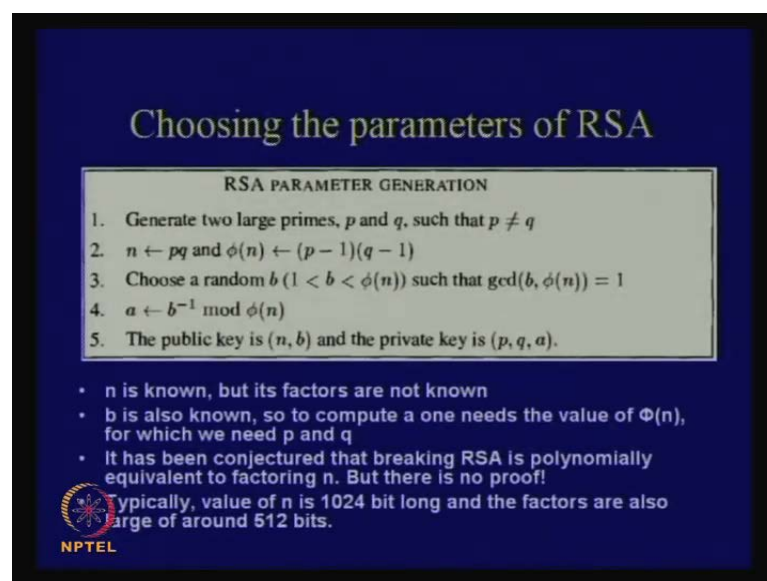
So, therefore, if you can compute phi  $n$ , that means what? That means that you can compute  $p$  plus  $q$ . If you can compute  $p$  plus  $q$ , you can also compute  $p$  minus  $q$  because you know  $p$   $q$ . So, that means you can solve this and you can find  $p$   $q$ . So, that means you can actually solve the factorization problem. So, but the thing is that, it is assume that factorization is quite a hard problem; again with 1,024 bits is really hard problem.

So, therefore, the idea is that, if you do not know the value of  $\phi(n)$ , from there computing the inverse of  $a$  is actually not so easy. So, that is again an assumption there is no proof. So, it is just to belief. So, therefore, I have written here that it has been conjectured that breaking RSA is polynomial equivalent factoring  $n$  but there is no proof, just to belief. So, typical value of  $n$  is generally 1,024 bit long and the factors are also large of around 512 bits.

So, here, you see that you have to do manipulation with so many bits. So, although you have got a good squarer multiply algorithm to do that but actually it is quite computationally intensive. So, although it is efficient in terms of complexity theory, but in terms of practical implementation, it is not really so efficient.

So, therefore, if I have a very handled device and implant an RSA algorithm into that, my power will get extinguish very soon. So, therefore, you see that when you need proper replacements of these kind of asymmetric key algorithms, and therefore, there are actually lot of work been done. For example, off let the one of the current standards to uses something which is called elliptic of cryptosystems. So, therefore, you can actually show that whatever security you get with say around 1,024 bits similar kind of security you can get with around say 192 bits of elliptic curves. So, therefore, that is quite efficient, but no doubt, RSA is wonderful algorithm.

(Refer Slide Time: 30:04)



**Choosing the parameters of RSA**

**RSA PARAMETER GENERATION**

1. Generate two large primes,  $p$  and  $q$ , such that  $p \neq q$
2.  $n \leftarrow pq$  and  $\phi(n) \leftarrow (p-1)(q-1)$
3. Choose a random  $b$  ( $1 < b < \phi(n)$ ) such that  $\text{gcd}(b, \phi(n)) = 1$
4.  $a \leftarrow b^{-1} \pmod{\phi(n)}$
5. The public key is  $(n, b)$  and the private key is  $(p, q, a)$ .

- $n$  is known, but its factors are not known
- $b$  is also known, so to compute  $a$  one needs the value of  $\phi(n)$ , for which we need  $p$  and  $q$
- It has been conjectured that breaking RSA is polynomially equivalent to factoring  $n$ . But there is no proof!

Typically, value of  $n$  is 1024 bit long and the factors are also large of around 512 bits.

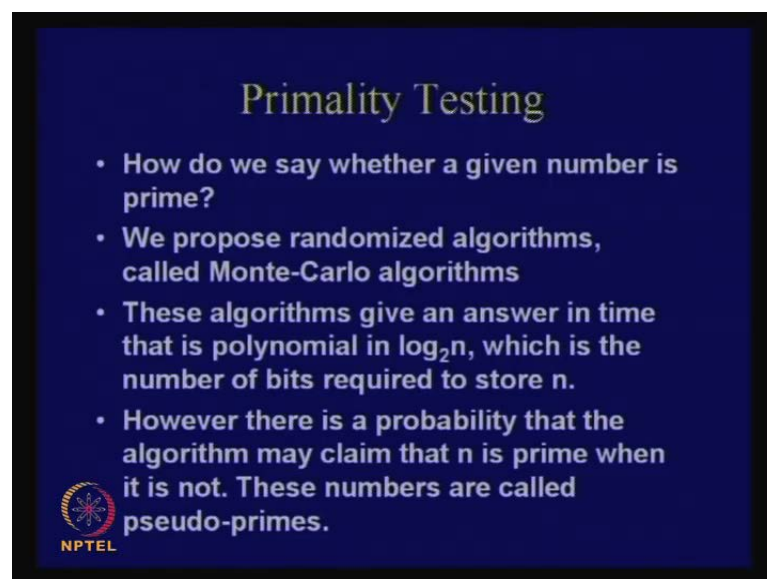
**NPTEL**

So, therefore, the next thing that we need to understand is that, we need to understand that whether, therefore, you see that the first step was to generate two large prime numbers -  $p$  and  $q$ . Therefore, how do you understand? If you I have got a large number say around 512 bits, you choose a number and you have to ascertain that whether that number is a prime number or not. So, that is also a quite difficult problem, and you must have heard of the phenomenal work which has been done by Agarwal and Kathiyala and Saxena the AKS algorithm which actually shows that you can actually solve this problem in polynomial time.

So what we will discuss is about some probabilistic methods of doing; some randomize algorithms through which we can actually approach is approach this problem. So, although that AKS problem is an undoubtedly a wonderful theoretical work, but the thing is that there are so many good probabilistic algorithms which has got a success probability very high and they are also very efficient.


So, therefore, this problem practically essentially is quite a solvable problem. Although I cannot say I mean without the AKS algorithm, it was not understood that you can actually said with certainty that it is a prime number, but actually you can using this algorithms, you can actually say that with a high probability. So, therefore, they Alice the implication of this, I mean why we need primary details. Therefore, given a number, you have to ascertain that whether the number is a prime number or not.

(Refer Slide Time: 31:30)



**Primality Testing**

- How do we say whether a given number is prime?
- We propose randomized algorithms, called Monte-Carlo algorithms
- These algorithms give an answer in time that is polynomial in  $\log_2 n$ , which is the number of bits required to store  $n$ .
- However there is a probability that the algorithm may claim that  $n$  is prime when it is not. These numbers are called pseudo-primes.

 NPTEL

So, therefore, the question that we address is that, how do we say whether a given number is prime or not, but we will actually not do it in only one class but we will go break that into two classes. So, therefore, but the thing is that first we need to understand certain things like; we will propose some randomized algorithms and these are called Monte Carlo algorithms.

So, what we have seen in the previous class was the Las Vegas algorithm and the difference between Las Vegas and Monte Carlo algorithm is this. That is in Las Vegas algorithms, what we saw was that it can fail, many, may be that it will not terminate, but if it terminates, it will give you a correct answer. But in case of Monte Carlo algorithms, it will definitely terminate, and there are two types of Monte Carlo algorithms - one which is called a yes based Monte Carlo algorithm and another which is called a no based Monte Carlo algorithm.

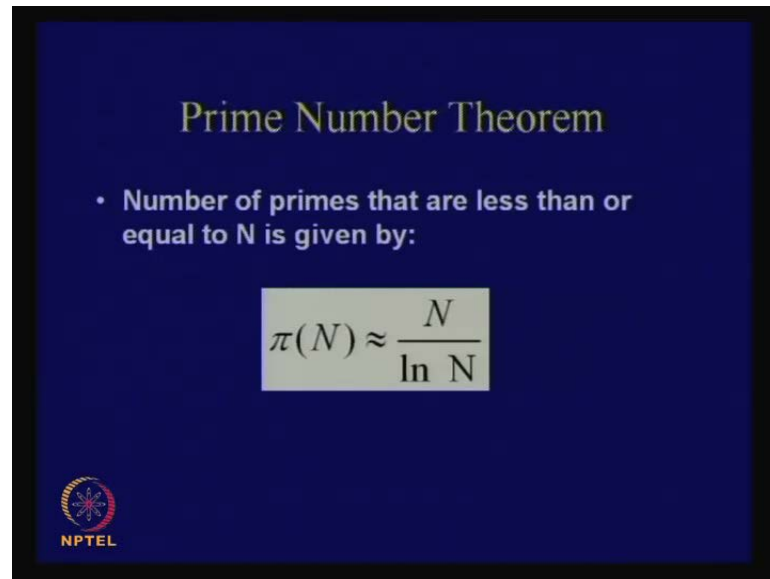
So, if it is a yes based Monte Carlo algorithm, then if it gives you an answer yes, it is correct; it is defiantly true, but if it says no, then there is an at most probability, like I mean that is an error probability in that. So, there is an estimate say epsilon which says the at most the error probability will be epsilon. So, that means that if the answer is yes, it is correct in case of yes based Monte Carlo algorithms.

But if the answer is no, then you have to take with a pinch of salt. So then, the idea is that, the idea is that there it is it is not defiantly trust worthy but there is an error maximum upper bound of the error probability. So, therefore, these algorithms will give you an answer during time that is polynomial, I mean, therefore, now come to this question, that is, we will engage some type of, we will see some types of Monte Carlo algorithm to do these and these algorithms actually given answered in time which is polynomial in say  $\log n$  base two. So, which is the number of bits required to store  $n$ ?

However, that is a probability that the algorithm may claim that  $n$  is prime when it is not. Therefore, it will say that some numbers are prime numbers where actually it is not a prime number and these numbers are called pseudo prime numbers. So, basically rather the problem that we will be addressing is actually whether it is composite or not. Therefore, it will be like given a number, you have to say whether it is composite or not. So, you understand it is a decisional problem yes or no.

But you do not, you are not solving the factorization problem. So, therefore, even if we have this polynomial time algorithm now like the AKS algorithm, RSA is not threatened because the RSA essentially believes or other lies upon assumption that you cannot factor. So, what we are only giving you is a primarily details but not a factorization algorithm follow.

(Refer Slide Time: 34:20)



Prime Number Theorem

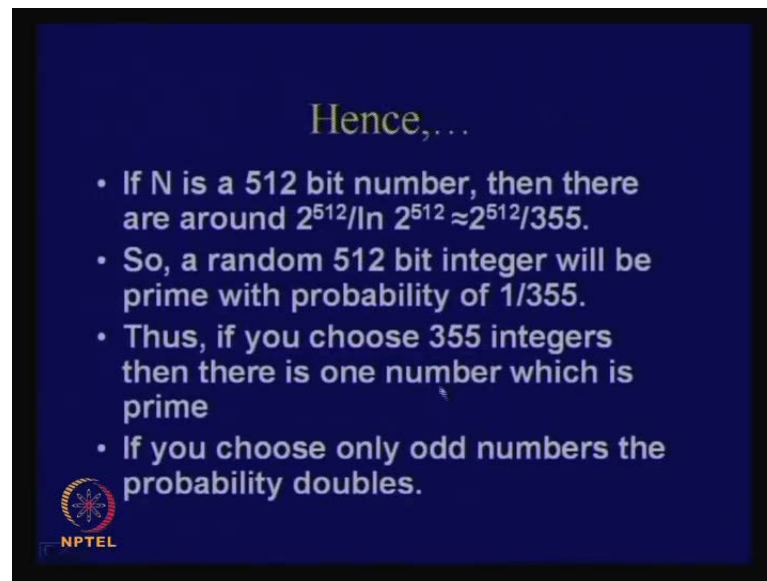
- Number of primes that are less than or equal to N is given by:

$$\pi(N) \approx \frac{N}{\ln N}$$

NPTEL

So, therefore, that is, there should be any confusion regarding that. So, we will just there is one theorem which is called the prime number theorem, which says that if the number primes less than or equal to N can be estimated by the formula, it says N by ln N. So, if you take say N is of the order of 512 bits, then N will be around 2 to the power of 512.

(Refer Slide Time: 34:43)



Hence,...

- If  $N$  is a 512 bit number, then there are around  $2^{512} / \ln 2^{512} \approx 2^{512} / 355$ .
- So, a random 512 bit integer will be prime with probability of  $1/355$ .
- Thus, if you choose 355 integers then there is one number which is prime
- If you choose only odd numbers the probability doubles.

NPTEL

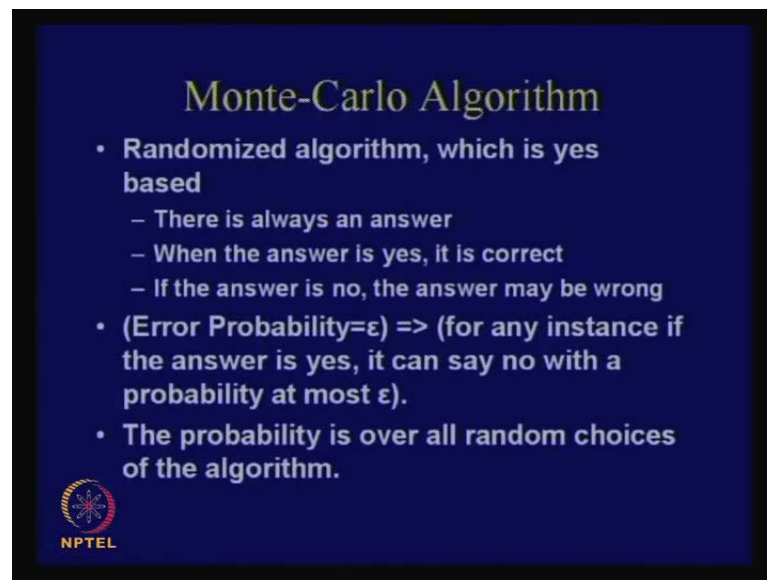
So, if you take  $\pi_n$  in that case or  $\pi$  to the power of 512, then it will work out to around 2 around 2 to the power of 512 divided by  $\ln 2$  to the power of 512. So, there is a problem here. So, it will around 2 to the power of, it will be around 2 to the power of 512 divided by 355. So, what you can show is that show a random. So, if you take a random 512 bit integer, so, if you take a random 512 bit integer, then it will be prime with a probability of around 1 by 355. So, what it shows is that, how many prime numbers do you have? So, it is around 2 to the power of 512 by 355, and how many numbers do you have? 2 to the power of 512.

So, the probability that a randomly chosen number is actually a prime number is actually as high as 1 by 355. So, which means if you repeat this, we have seen now that, if there is an experiment which is a probability of 1 by  $p$ , if I repeat that experiment for  $p$  times, then we are bound to get one success. The expectation is that we will get one, one, success then the probability of that is quite high.

So, therefore, if I repeat this experiment for 355 times, then there is a high probability that I will get at least one prime number. So, that number is not so large you know. So, I can do that, and actually if we choose only odd numbers, then you can actually double your probability that, instead of randomly choosing numbers, I only choose the odd integers. So, that will double my probability.


Yes, that will, therefore, yes, therefore, the probability of this is quite high, and therefore, what we will see is that the expected number of trials is not so large. Therefore, I can engage randomized algorithms to that; random algorithms which are just based upon some random decisions.

(Refer Slide Time: 36:38)



**Monte-Carlo Algorithm**

- Randomized algorithm, which is yes based
  - There is always an answer
  - When the answer is yes, it is correct
  - If the answer is no, the answer may be wrong
- (Error Probability= $\epsilon$ ) => (for any instance if the answer is yes, it can say no with a probability at most  $\epsilon$ ).
- The probability is over all random choices of the algorithm.

 NPTEL

So, now, let us see what is a Monte Carlo algorithm. So, idea is that these are randomized algorithms, which is, which are generally yes based. So, it is yes based. Therefore, it means that there is always an answer. When the answer is yes, it is correct, and if the answer is no, then the answer may be wrong.

And if I say that error probability of an yes based Monte Carlo algorithm is epsilon, then I mean to say that for any instance if the answer is yes, it can say no with a probability of at most epsilon. So, therefore, for any instance if the answer is yes, it can say no with a probability of at most epsilon. So, therefore, this probability is again thus we have seen in case of Las Vegas algorithms also is over all the random choices of the algorithm; that means if the algorithm receive some inputs, then it is over the input space.



(Refer Slide Time: 37:34)

The slide is titled "The Problem Composites" in a serif font. Below the title is a light-colored box containing the following text:

**Composites**  
**Instance:** A positive integer  $n \geq 2$ .  
**Question:** Is  $n$  composite?

Below the box, there is a bulleted list of four points:

- This is a decision problem.
- We will discuss the Solovay-Strassen Algorithm, which is a Monte-Carlo algorithm for Composites.
- Thus if it says yes,  $n$  is surely composite.
- However, if  $n$  is composite then it says yes with probability at least  $\frac{1}{2}$

In the bottom left corner of the slide, there is a small circular logo with a red and blue design, and the text "NPTEL" below it.

So, the question that will be addressing is this, that is, given a positive integer  $n$  greater than or equal to two is  $n$  composite. So, you understand that is a decisional problem and we will try to discuss some an algorithm which is called the Solovay Strassen algorithm to solve this and it is a Monte Carlo algorithm for composites. Therefore, it says yes. If it say yes, then  $n$  is really composite because the problem is composite; it is composite, so, which says yes. So, first thing is it will defiantly terminate. Second thing is that if it says yes, then it is correct but it can say no also. If it says no, then a probability of error is at most half.

(Refer Slide Time: 38:19)

$n = pq$   
 $\phi(n) = (p-1)(q-1)$   
 $= pq - (p+q) + 1$

Composite  
├── yes  
└── No.

$\Pr[\text{No} \mid \text{Composite}] \leq \frac{1}{2}$   
 $\Rightarrow \Pr[\text{Yes} \mid \text{Composite}] \geq \frac{1}{2}$

NPTEL

So, in that case, what I am saying is like this, that is, if it is composite, there can be two cases - one is yes and the other one is no. So, if it is yes, it is defiantly composite, but what I am saying is that, the, if the probability, if it, I mean given, then it is composite the probability that it will say you a no is actually at, I mean the probability of it is, so, it is an error. So, therefore, this will be at most half. So, it says probability of no given composite is at most half. Therefore, it is less than or equal to half; so, that means that probability that if I say you a yes, given it is composite is greater than equal to half.


So, therefore, that means that, if the number  $n$  is composite, then it says yes with a probability of at least half, but to understand that, again we need some more number theoretic results and that is something which is called quadratic residues.

(Refer Slide Time: 39:37)

## Quadratic Residue

Suppose  $p$  is an odd prime and  $a$  is an integer.  $a$  is defined to be a *quadratic residue* modulo  $p$  if  $a \not\equiv 0 \pmod{p}$  and the congruence  $y^2 \equiv a \pmod{p}$  has a solution  $y \in \mathbb{Z}_p$ .  $a$  is defined to be a *quadratic non-residue* modulo  $p$  if  $a \not\equiv 0 \pmod{p}$  and  $a$  is not a quadratic residue modulo  $p$ .

- There are exactly  $(p-1)/2$  QR (Quadratic Residues)




So, the idea of quadratic residues is quite nice and it is quite easy also to understand. So, what is this, that is, suppose, so, the let us see the definition. So, definition is as follows, that is, suppose  $p$  is an odd prime number. Therefore,  $p$  is an odd prime number means what? Exclude 2 and  $a$  is an integer. Then  $a$  is define to be a quadratic residue modulo  $p$  if  $a$  is not congruent to  $0 \pmod{p}$  and the congruence  $y^2 \equiv a \pmod{p}$  has a solution  $y$  which lies inside  $\mathbb{Z}$ .

(Refer Slide Time: 40:33)

© IIT KGP

a QR .  $\forall a \not\equiv 0 \pmod{p}$ .

$y^2 \equiv a \pmod{p} \Rightarrow y \in \mathbb{Z}_p$ .



So, therefore, the equation  $y^2 \equiv a \pmod{p}$  will have a solution for  $y$  which belongs to  $\mathbb{Z}_p$ . So, therefore, if I will say you that there is a quadratic residue and if I say you that  $a$  is a quadratic residue, that means two things; that means first it means that  $a$  is not congruent to  $0 \pmod{p}$ . So, what does it mean?  $a$  does not divide  $p$ . So,  $a$  is co-prime to  $p$  basically. Therefore, so,  $a$  is not congruent to  $0 \pmod{p}$ , and the other thing is that, I mean for if there is an equation of  $y^2 \equiv a \pmod{p}$ , then there is a solution for  $y$  where  $y$  belongs to  $\mathbb{Z}_p$ .

(Refer Slide Time: 41:15)

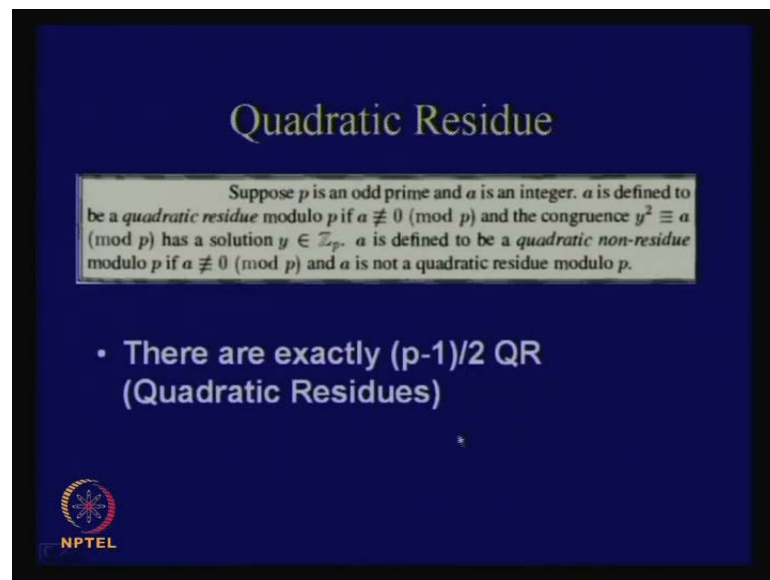
The slide is titled "Example" and features a list of squares in  $\mathbb{Z}_{11}$  on the left and a note in a grey box on the right. The list shows that the quadratic residues form a palindromic sequence:  $1^2=1, 2^2=4, 3^2=9, 4^2=5, 5^2=3, 6^2=3, 7^2=5, 8^2=9, 9^2=4, 10^2=1$ . The note states: "Note, that the QR forms a palindrome. There are exactly  $(11-1)/2=5$  QRs." The NPTEL logo is visible in the bottom left corner of the slide.

So, let see one example. So, example would be like this say in  $\mathbb{Z}_{11}$ , consider  $\mathbb{Z}_{11}$  and find out  $1^2, 2^2, 3^2, 4^2$  and find out all the squares of that. So, that means that  $1^2 \equiv 1$ ,  $2^2 \equiv 4$ ,  $3^2 \equiv 9$ ,  $4^2 \equiv 5$ . So, what does it mean that, you see that all these values are essentially the quadratic residues.

Because if I give you  $5$ , then you can find out a  $4$  which is, and if you raise it to the power of  $2$ , you get obtain  $5$ . See, you see that this is a palindromic sequence. You get a  $1$  here; you get a  $1$  here; you get a  $4$  here; you get a  $4$  here. Why? It is quite obvious, why, because you can actually write this  $10$  as  $11 - 1$ , and if you take modulo  $11$ , whether you take a square, all those  $11$ , I mean all other terms of multiples of  $11$ . Therefore, they vanish and you get back only minus  $1$  square which is  $1$ .

So, therefore, in this fashion, you can understand that you will always find, I mean that the number of quadratic residues in this case will be exactly equal to 11 minus 1 by 2, because of this, because you will always have an even number, 1 here, 1 here, 4 here, 4 here. So, you can find out how many number of quadratic residues will be there.

(Refer Slide Time: 42:34)



So, if you generalize this, then actually there are exactly  $p$  minus 1 by 2 quadratic residues. So, if this is not the, not the case, that is  $a$  is define to be a quadratic non-residue modulo  $p$ . If  $a$  is not congruent to 0 modulo  $p$  and  $a$  is not a quadratic residue modulo  $p$ .

So, in this case, what are quadratic residues? See, for example, 6 is a non-quadratic residue; 2 is a non-quadratic residue. So, how many non-quadratic residue are there, that is also same; that is also 5 because we have excluded the  $n$ . We have said at beginning only, I mean  $a$  is not congruent to 0 mod  $p$ . If we see the definition, in both quadric residues or non-quadratic residues, this  $a$  congruent to 0 mod  $p$  is excluded.

(C)

(Refer Slide Time: 43:59)

**Generalization**

How many solutions are there to  $x^2 = a(\text{mod } p)$  for odd positive prime  $p$ ?

If,  $y^2 = a(\text{mod } p), y \in Z_p^*$

then  $(-y)^2 = a(\text{mod } p)$

Note,  $y = -y(\text{mod } p)$ , as  $p$  is odd

Thus, the quadratic congruence:

$$x^2 - a = 0(\text{mod } p)$$


can be factored into

$$(x - y)(x + y) = 0(\text{mod } p)$$

Since,  $p$  is prime,  $p \mid (x - y)$  or  $p \mid (x + y)$

Thus,  $x = \pm y(\text{mod } p)$

Thus, there are exactly two solutions of the congruence.



Is it so? Anyone yet? How many numbers are there in  $Z_{11}$ ? There are eleven numbers. So, that is exactly same. So, therefore, how many solution are there to this question  $x$  square equal to  $a$  mod  $p$ ? So, you can actually prove that, I mean it is quite easy also. You have understood more or less the idea, then it will be exactly there are two solutions for the congruence.


(Refer Slide Time: 44:19)

**Example**

- $Z_{11}$
- $1^2=1$
- $2^2=4$
- $3^2=9$
- $4^2=5$
- $5^2=3$
- $6^2=3$
- $7^2=5$
- $8^2=9$
- $9^2=4$
- $10^2=1$

Note, that the QR forms a palindrome

There are exactly  $(11-1)/2=5$  QRs.



So, therefore, if I give you one value here, say if I give you 1, then how many values from where you can get 1? You can get from 1 and you can also get from 10. So, for

every number  $a$ , there can be two wise. If you square, you will get back  $a$  and the proof is exactly same as what we thought it is quite simple.

Only I mean if I go through details, then minus it says that, if  $y$  squared is congruent to  $a$  mod  $p$ , then  $y$  belongs to  $Z$  and  $y$  belongs to  $Z_p^*$ . Then minus  $y$  whole square is also equal to  $a$ . Therefore, you can actually understand easily that  $y$  and minus  $y$  are both the solutions. So,  $y$  and minus  $1$  means in this case  $p$  minus  $y$ .

So, one solution is  $y$  and the other one is  $p$  minus  $y$ . If you square them, you will get back  $a$ , but why are there exactly two results, because if you write like  $x$  square minus  $a$  equal to you know I am congruent to  $0$  mod  $p$ , then you can factor that and it will become equal to  $x$  minus  $y$  into  $x$  plus  $y$  congruent to  $0$  modulo  $p$  and  $c$  is prime  $p$  is prime, then  $p$  divides  $x$  minus  $p$  or  $p$  divides  $x$  plus  $y$ .

(Refer Slide Time: 45:31)

The QR Problem

|                    |   |
|--------------------|---|
| Quadratic Residues |   |
| Instance:          | An odd prime $p$ , and an integer $a$ . |
| Question:          | Is $a$ a quadratic residue modulo $p$ ? |

- We have a polynomial time deterministic algorithm to solve this decision problem.

NPTEL

So, that means  $x$  is congruent to plus minus  $y$  mod  $p$ . So, there are exactly two solutions which will satisfy this equation. So, the proof is exactly intuitively what we understand, but what is the quadratic residue problem. Therefore, how do you understand that whether a number is quadratic residue? So, quadratic residue what you can do is that, the problem is this an odd prime  $p$  and an integer  $a$  is given and the question is a a quadratic residue modulo  $p$ .


(Refer Slide Time: 46:00)

**Euler comes to the rescue again**

*(Euler's Criterion) Let  $p$  be an odd prime. Then  $a$  is a quadratic residue modulo  $p$  if and only if*

$$a^{(p-1)/2} \equiv 1 \pmod{p}.$$

- The time complexity of this check is  $O(\log p)^3$  by applying square and multiply method to raise an element to a power.
- Note that if  $a^{(p-1)/2} \equiv -1 \pmod{p}$  then  $a$  is a non-quadratic residue.

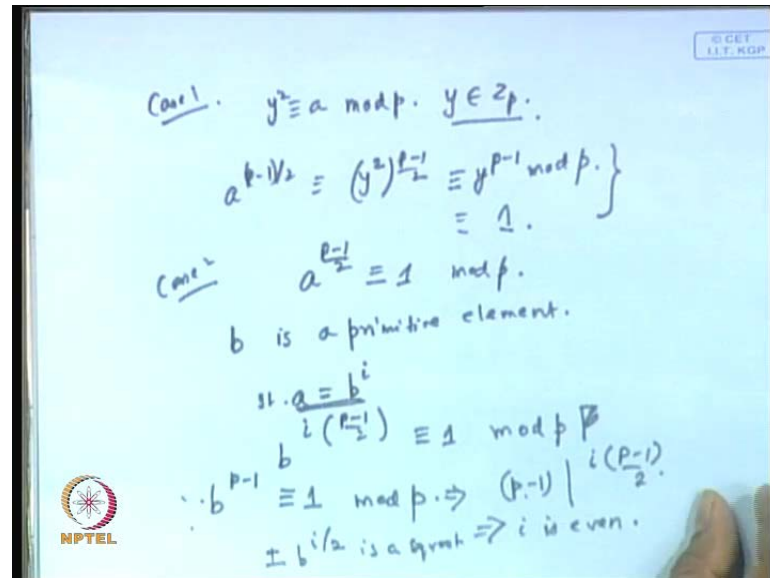


So, what you can do is, first of all, you can raise and find out like what we have done for  $Z_{11}$ , but can there be any better algorithm than that? So, in order to solve that, again all are come to the rescue and we you have got this nice theorem. It says that let  $p$  be an odd prime number, then  $a$  is a quadratic residue modulo  $p$  if and only if this is satisfied, that is,  $a$  to the power of  $p$  minus 1 by 2 is congruent to 1 modulo  $p$ .

And the time complexity of this check is quite easy to understand is whole  $\log p$  whole cube, because what you have to do is that you have to raise this power raise, and raising power you know is  $O(\log p)$  whole cube, why, because if you just think of the square and multiply algorithm, do you understand why it is  $O(\log p)$  whole cube?



(Refer Slide Time: 46:51)



So, can you, can you, can you justify why it is true? So, you see that there are two parts if and only if. So, let us start 1 by 1, I mean it is quite easy with the proof. So, the case one will be like, if you say that a is a quadratic residue modulo p, so, that means what? That is, there is a solution y if you raise to the power of a.

So, y lies insight  $Z_p$ . So, there have to exist such a y. Now, what about the equation a to the power of p minus 1 by 2 in in that case? It is y to the power of two whole to the power p minus 1 by 2. So, what is that equal to y to the power of p minus 1 mod p, and what is that?

Y to the power of  $(\ )$

Yeah, what is that equal to?

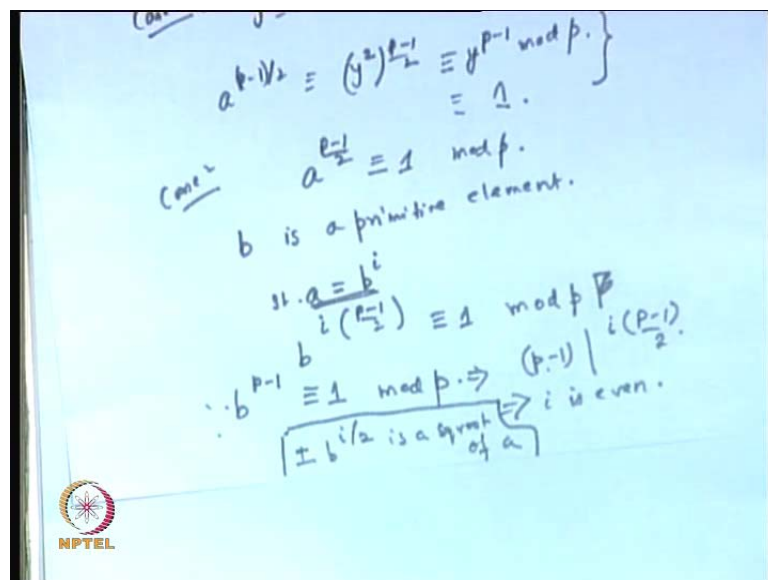
$(\ )$

Equal to 1. That is the form Euler's little theorem. So, y to the power of p minus 1 mod p is equal to 1. Therefore, this proves the only if part. What about the other part? You start with those, that is, a to the power of p minus 1 by 2 is equal to 1 mod p, and by my previous days, you will remember that cyclic group idea. Then actually I can, if there is an elements say b which is a primitive element in this group, there has to exist one primitive element.

So, if I say that  $b$  is the primitive element, then I can always write  $a$  as  $b$  to the power of some  $I$  value, and you know what is the range of  $I$ . So, it has to lie inside that cyclic group. So, therefore, in that case, if you plug in  $b$  to the power of  $I$  here, you get  $b$  to the power into  $p$  minus 1 by 2 is congruent to 1 mod  $p$ , but we know that  $b$  to the power of  $p$  minus 1 is equal to 1 mod  $p$  and  $p$  minus 1 is the minimum is the least number which is like that, because why? Because  $b$  is a primitive element. It is a least number for which this equation is obtained.

So, that means that  $p$  minus 1 has to divide  $I$  into  $p$  minus 1 by 2 because  $p$  minus 1 by  $I$  into  $p$  minus 1 by 2 has to be a bigger number. So, if,  $I$ ,  $p$  minus 1 divides  $I$  into  $p$  minus 1 by 2, it means that  $I$  is divided by 2, that is,  $I$  is an even number.

(Refer Slide Time: 49:26)



So, this implies that  $I$  is even. So, if  $I$  is even, then you see that from this equation, you can obtain a square root of this  $a$ , and therefore, you can say that plus minus  $b$  to the power of  $I$  by 2 is a square root of  $a$ , and that proves that  $a$  is quadratic residue. So, we stop at this point and we will continue. We will see that how we can use this to understand the primality test, I mean, yeah, to understand the primality test and other certain other issues. So, again follow that up with the factorization problem also. You stop at this point.