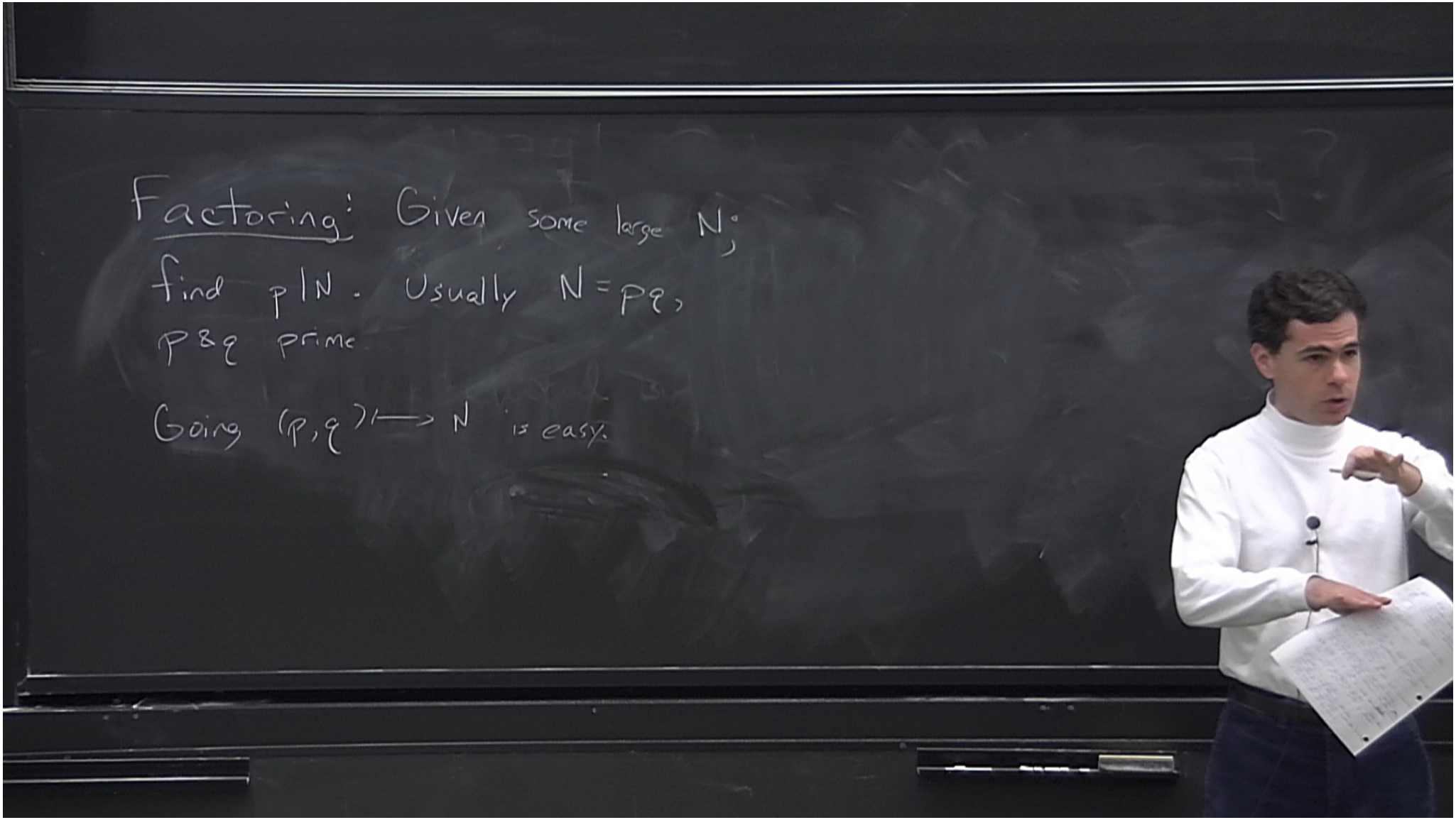
Title: Quantum Information (Review) - Lecture 7

Date: Feb 22, 2012 10:15 AM

URL: http://pirsa.org/12020039

Abstract:


Factoring: Given some large N;
find p|N. Usually N = pq,
p & q prime
Going (p,q) ⟼ N is easy.

Factoring: Given some large N;
find $p \mid N$. Usually $N = pq$,
p & q prime.

Going $(p, q) \longmapsto N$ is easy.
Time $O((\log N)^2)$

: Given some large N;

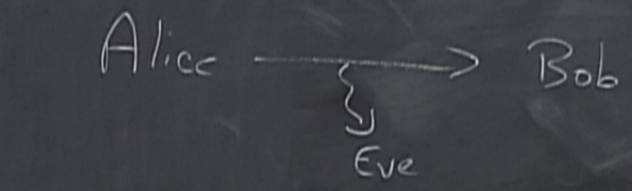
- Usually $N = pq$,

$\longmapsto N$ is easy.

$((\log N)^2)$

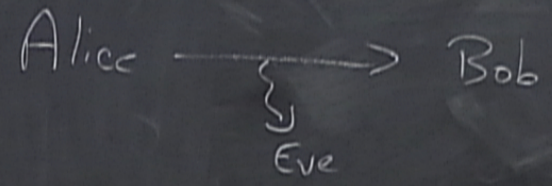
RSA public key encryption:

RSA public key encryption:

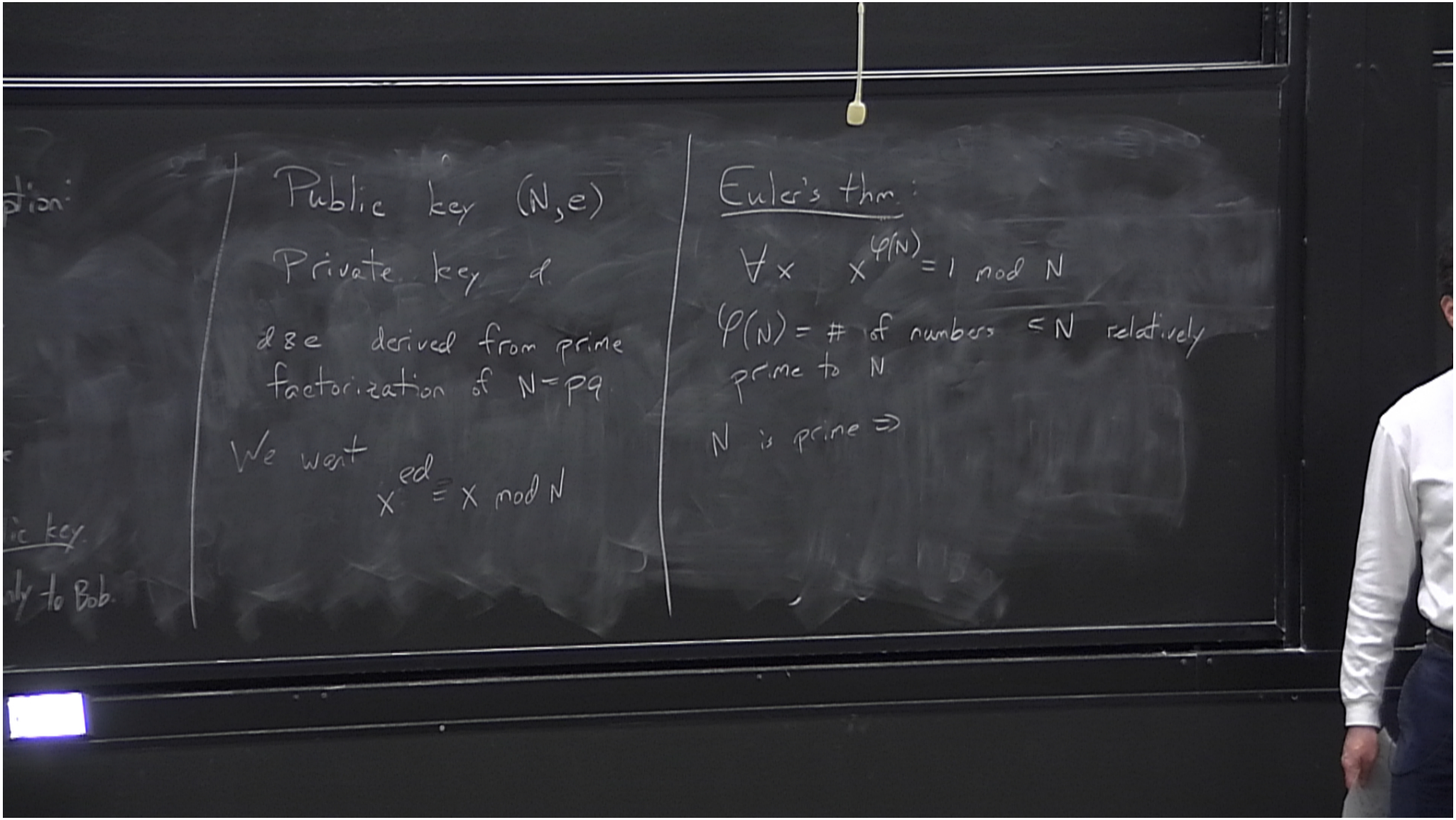


Bob should be able to decode
Eve should not
Encryption performed with public key

RSA public key encryption:



Bob should be able to decode
Eve should not
Encryption performed with public key
Decrypt using private key, known only to Bob.


Public key (N, e)
Private key d
d & e derived from prime factorization of N = pq
We want
$x^{ed} \equiv x \bmod N$
Euler's thm:
$\forall x \quad x^{\varphi(N)} = 1 \bmod N$
$\varphi(N)$ = # of numbers $\leq N$ relatively prime to N
N is prime ⇒

tion:

ic key

ly to Bob.

Public key (N, e)

Private key d

d & e derived from prime factorization of $N = pq$

We want

$$x^{ed} \equiv x \bmod N$$

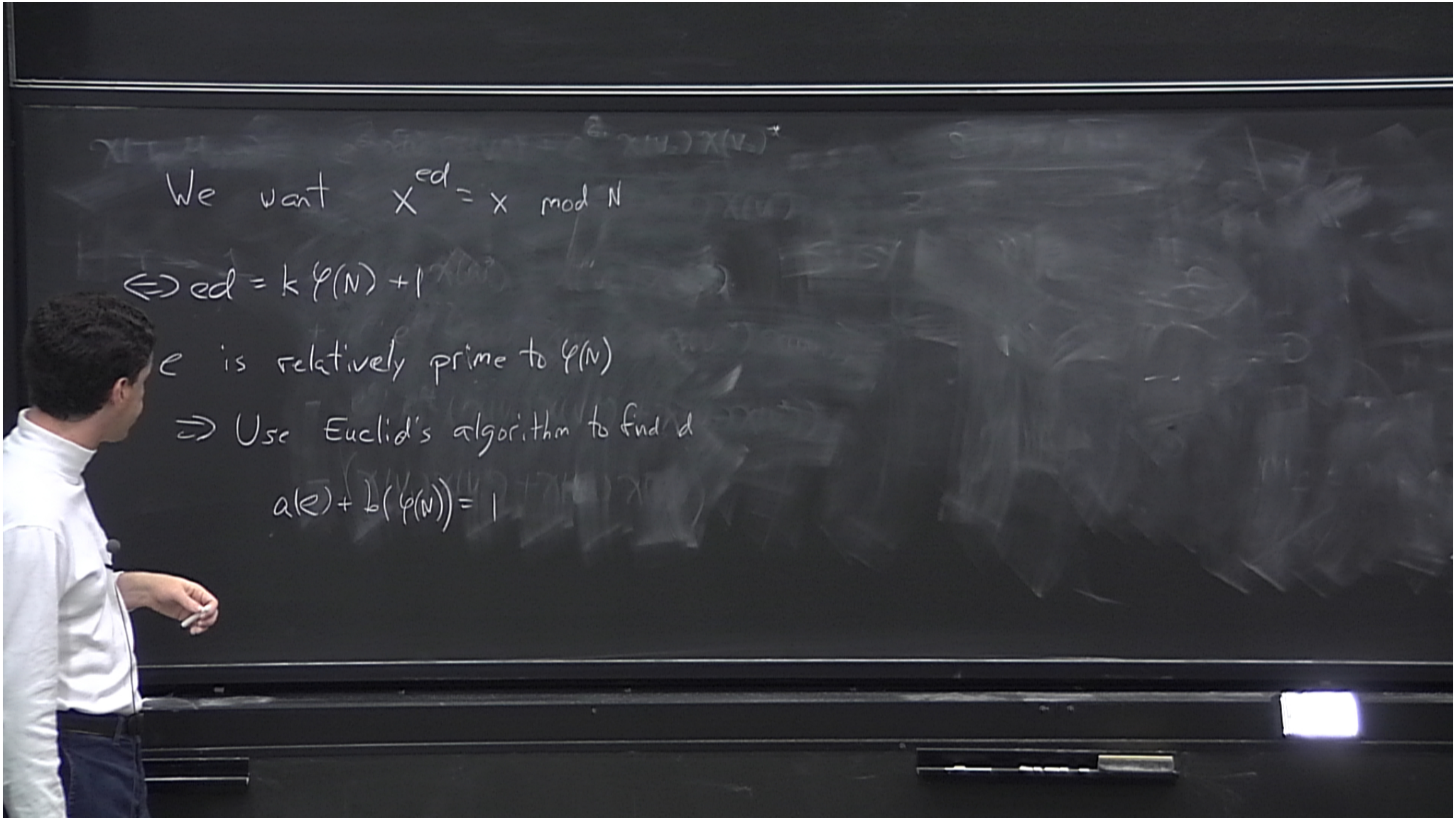
Euler's thm:

$$\forall x \quad x^{\varphi(N)} = 1 \bmod N$$

$\varphi(N)$ = # of numbers $\leq N$ relatively prime to N

N is prime $\Rightarrow \varphi(N) = N - 1$

$N = pq \Rightarrow \varphi(N) = (p-1)(q-1)$


We want $x^{ed} = x \mod N$
$\Leftrightarrow ed = k\varphi(N) + 1$
e is relatively prime to $\varphi(N)$
$\Rightarrow$ Use Euclid's algorithm to find d
$a(e) + b(\varphi(N)) = 1$

Going $(p,q) \longmapsto N$ is easy.

Time $O((\log N)^2)$

Eve

Bob should be able to decode

Eve should not

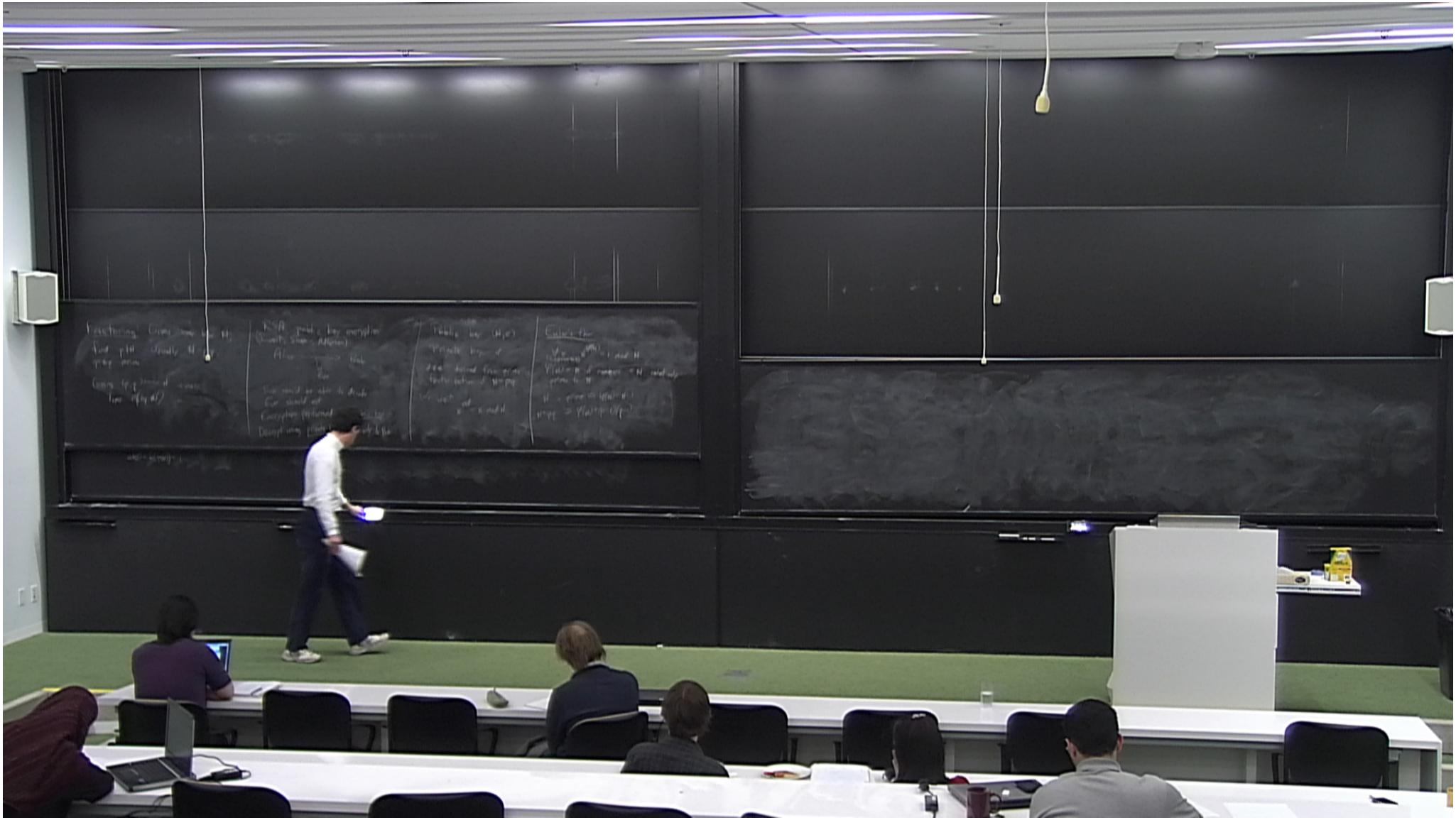
Encryption performed with public key

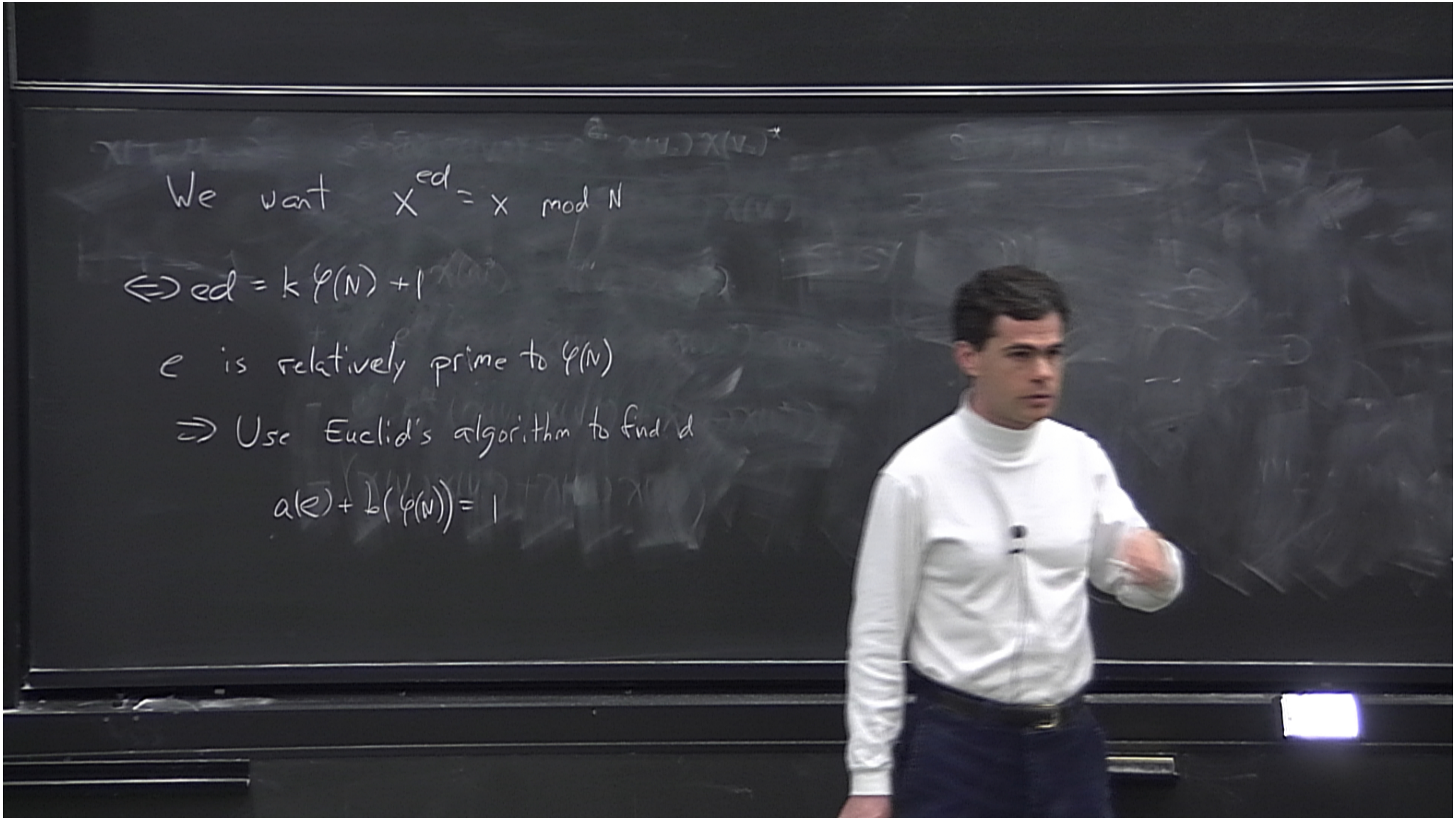
Decrypt using private key, known only to Bob.

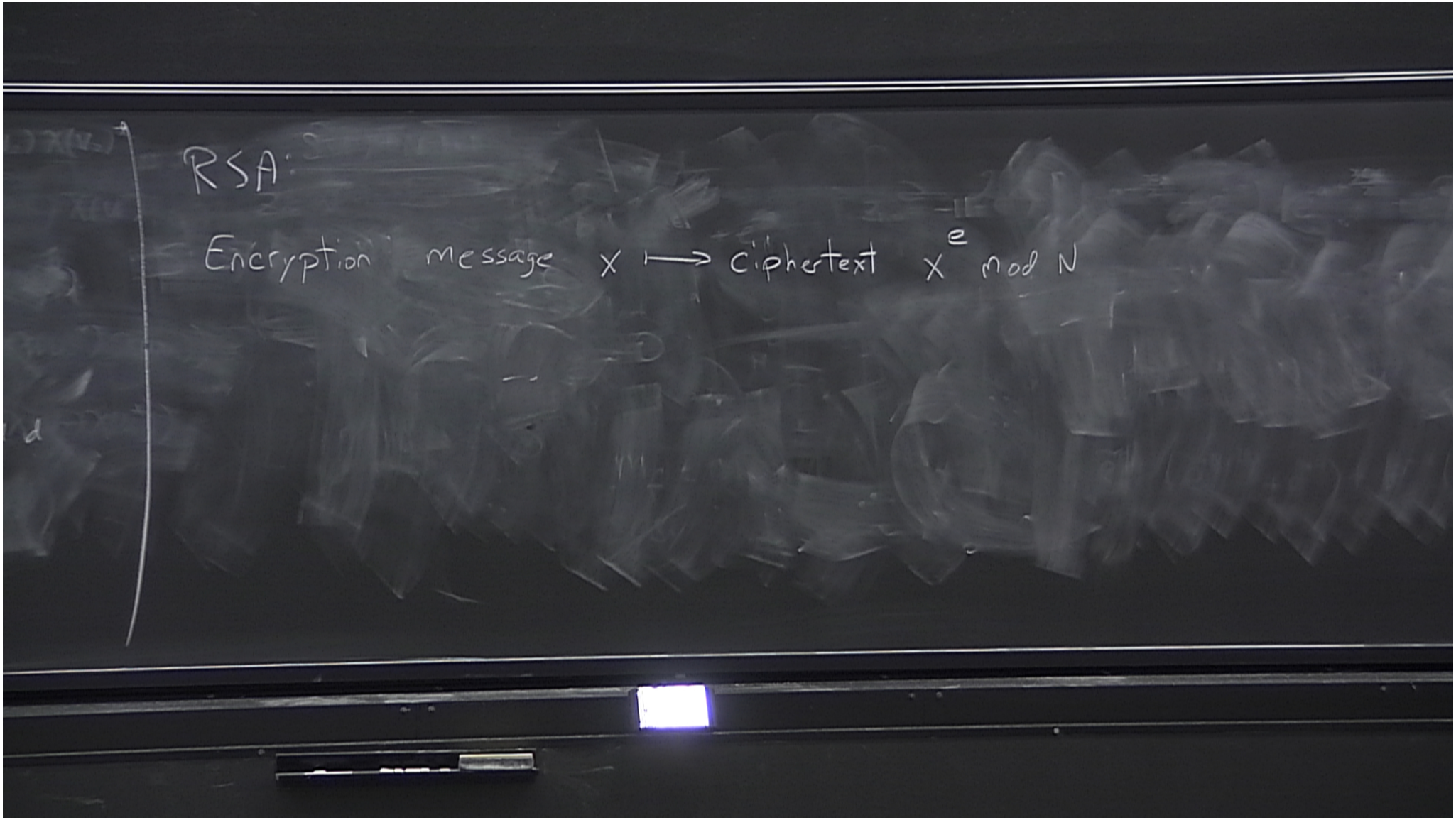
e is relatively prime to $\varphi(N)$

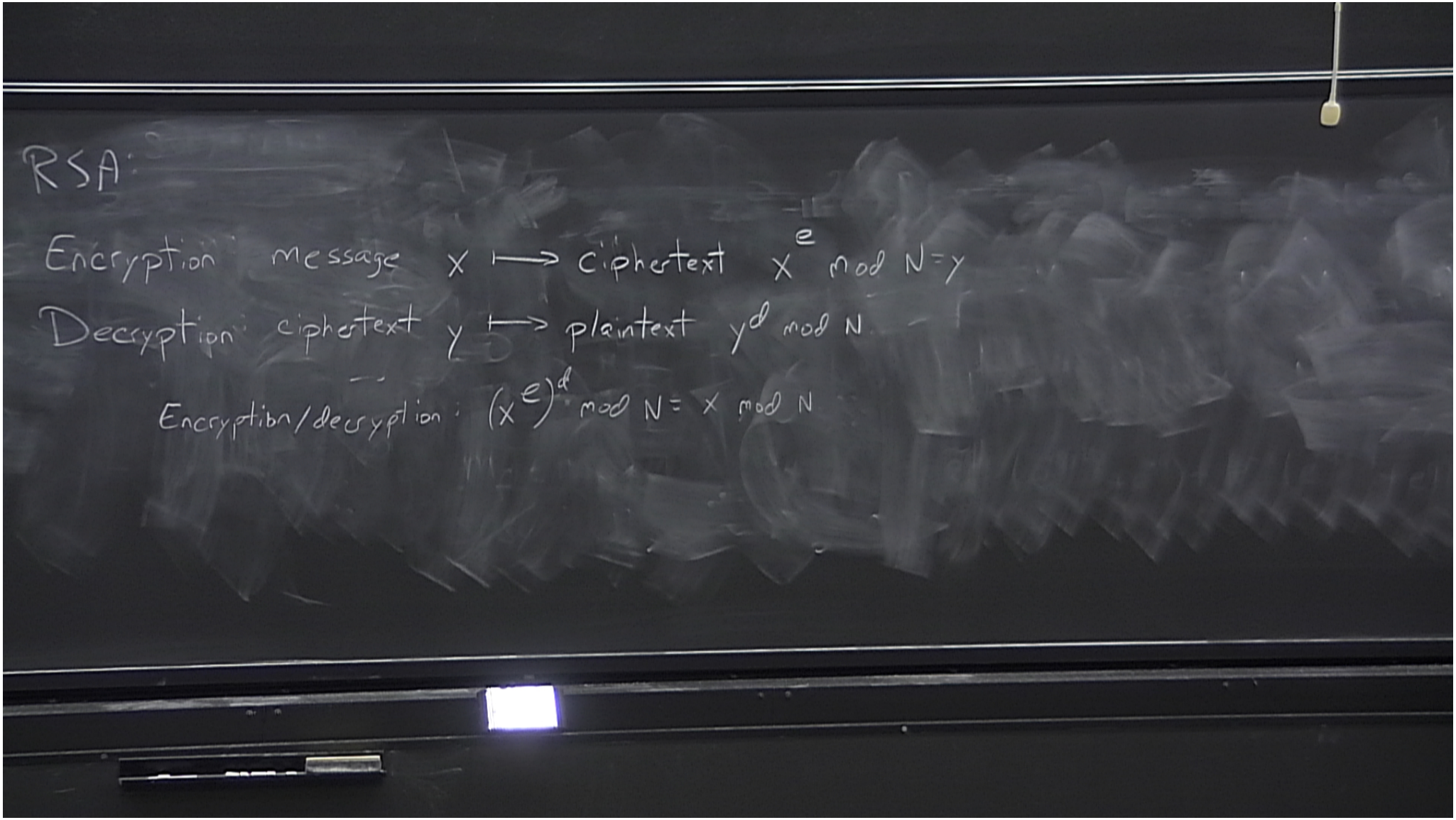
$\Rightarrow$ Use Euclid's algorithm to find d

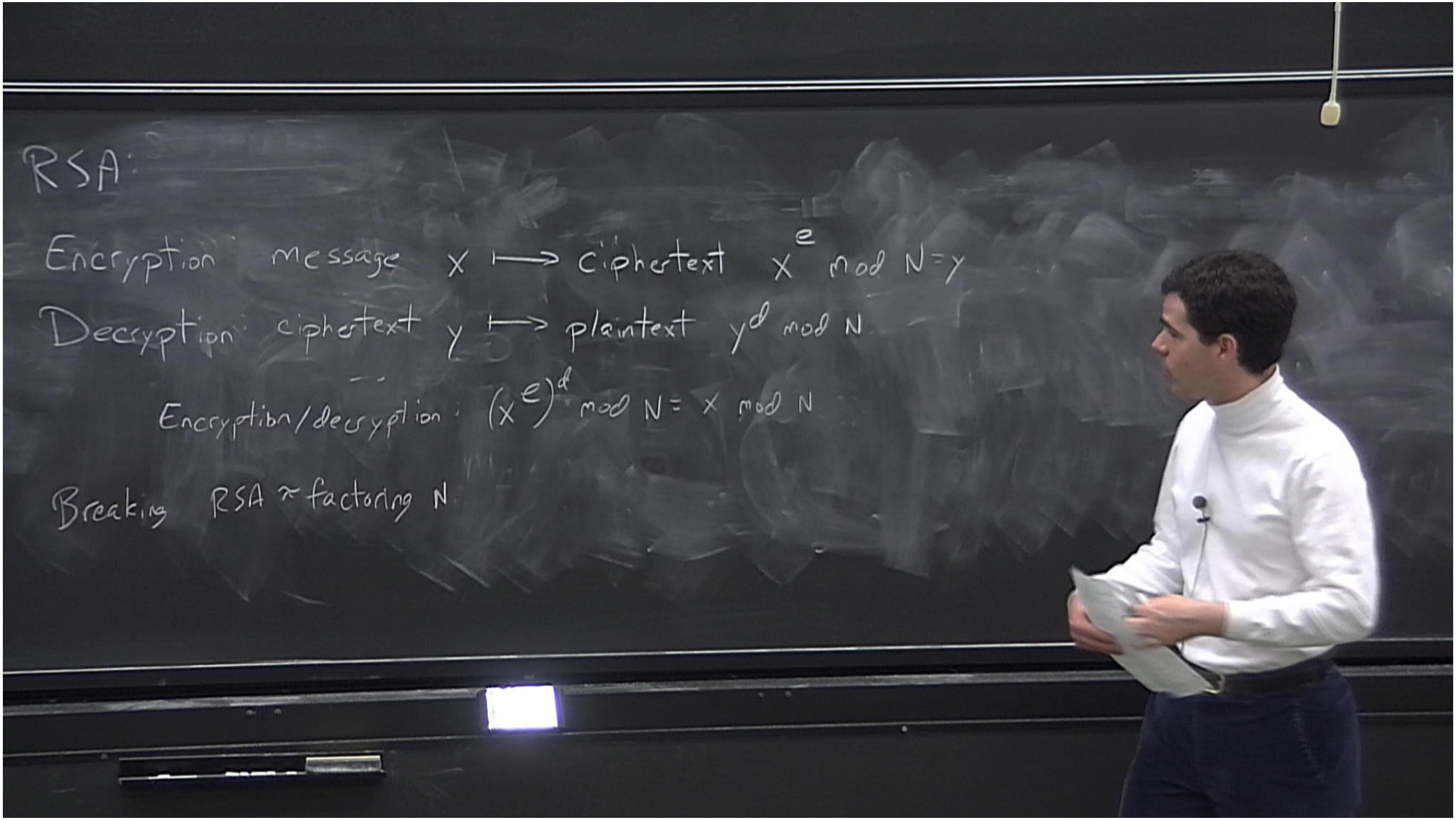
$$a(e) + b(\varphi(N)) = 1$$

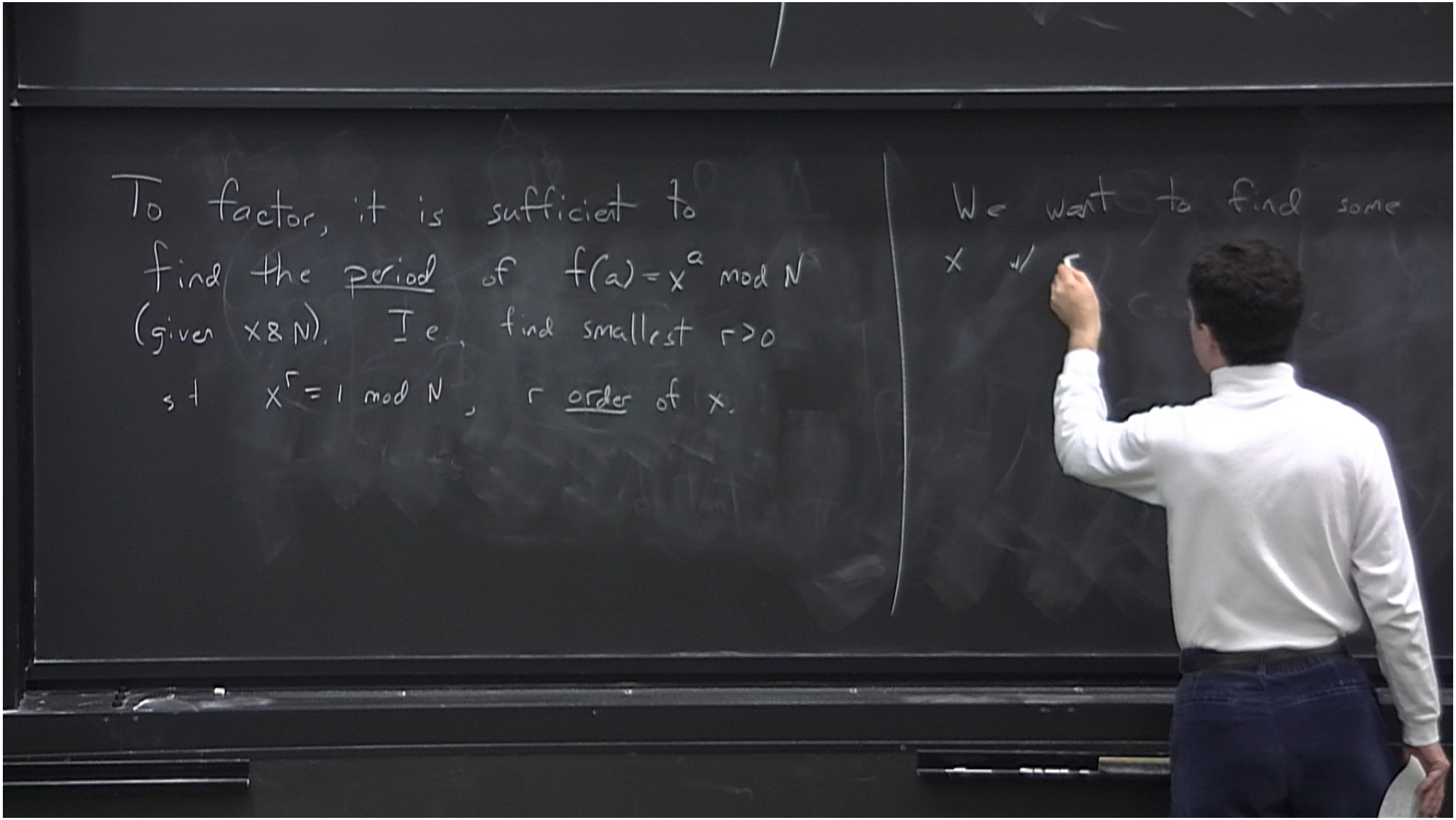


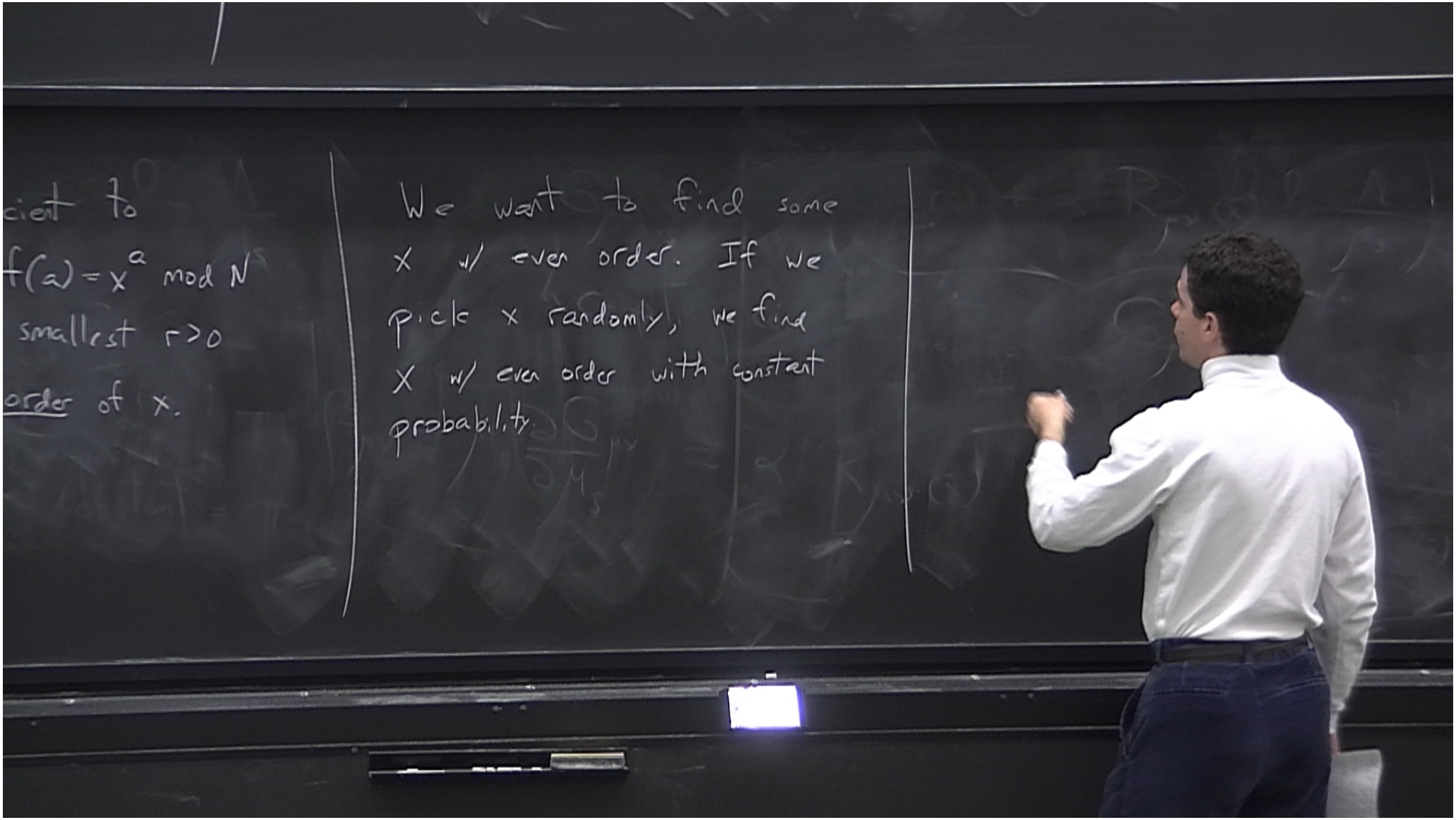

We want $x^{ed} = x \mod N$
$\Leftrightarrow ed = k\varphi(N) + 1$
e is relatively prime to $\varphi(N)$
$\Rightarrow$ Use Euclid's algorithm to find d
$a(e) + b(\varphi(N)) = 1$

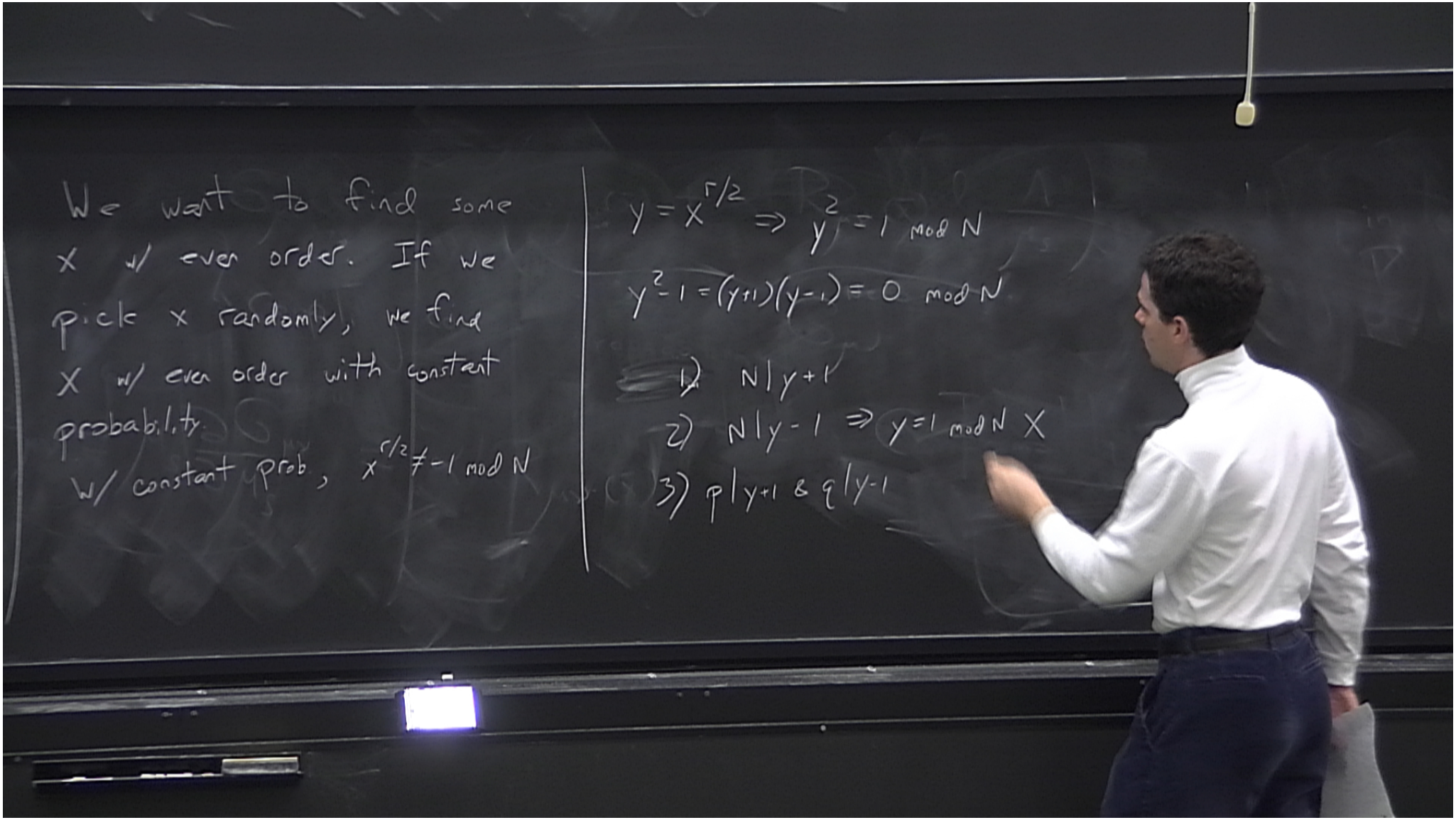

RSA:
Encryption message $X \longmapsto$ ciphertext $X^e \bmod N$

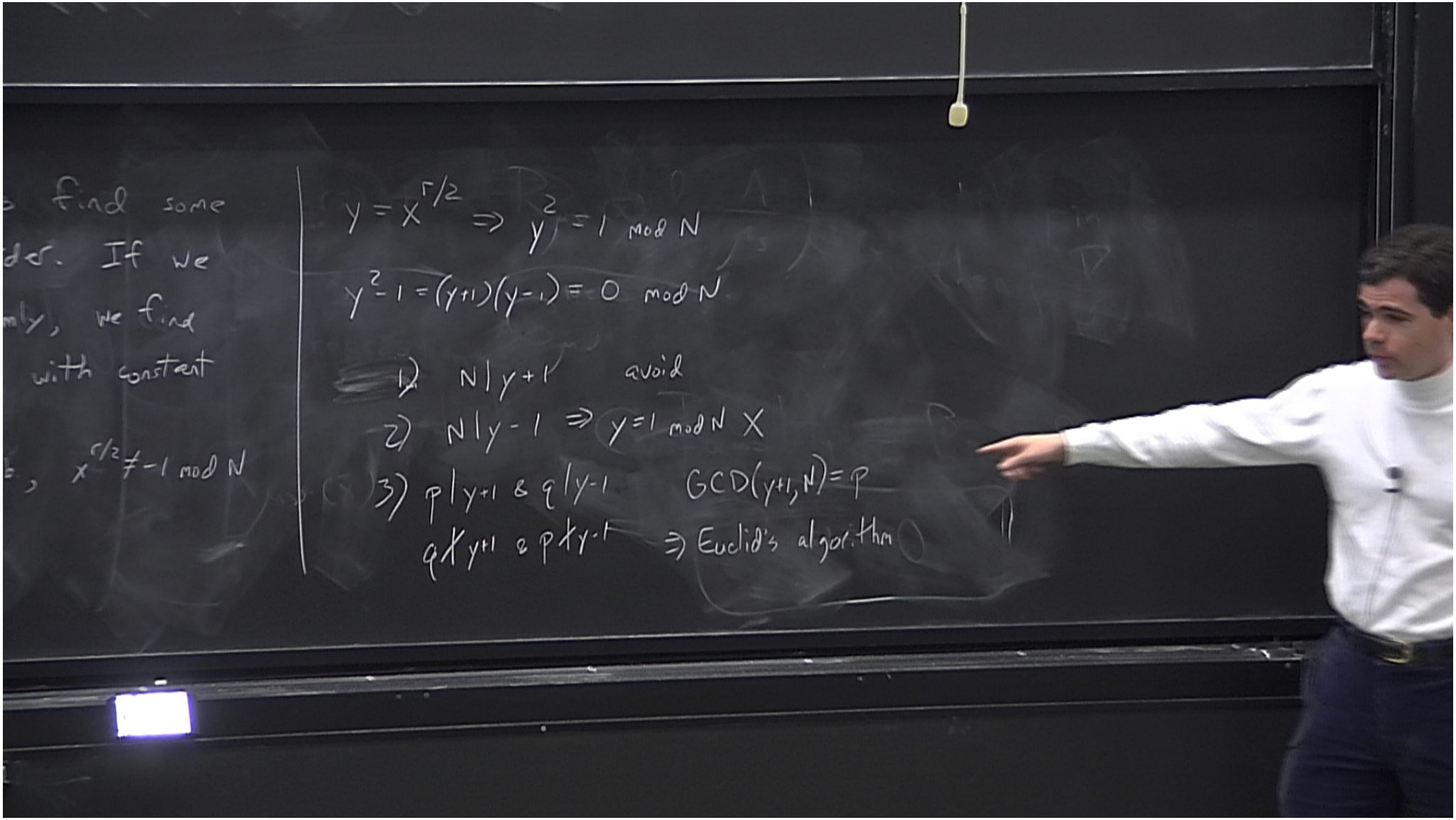

RSA:
Encryption: message $x \longmapsto$ ciphertext $x^e \bmod N = y$
Decryption: ciphertext $y \longmapsto$ plaintext $y^d \bmod N$
Encryption/decryption: $(x^e)^d \bmod N = x \bmod N$

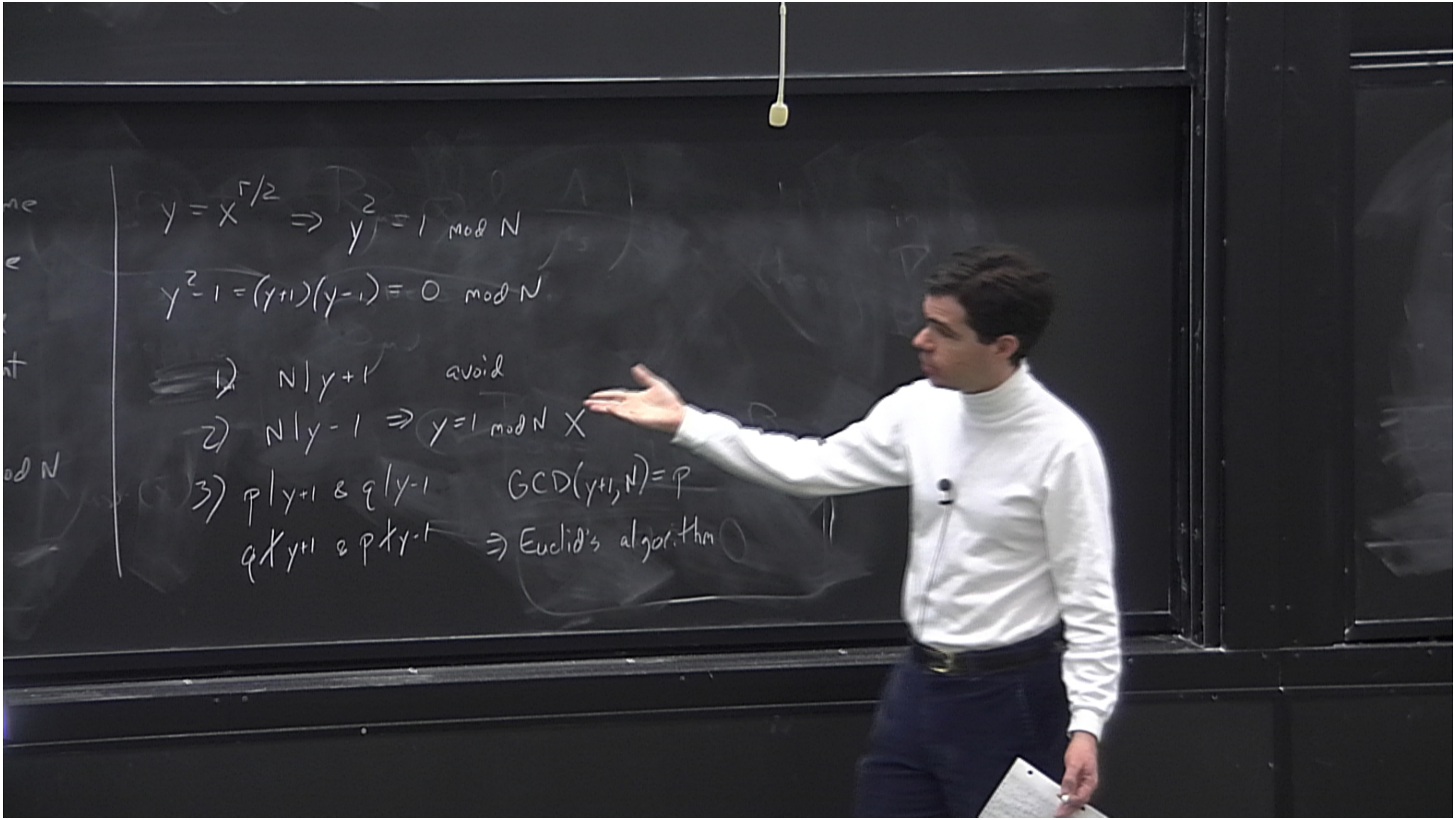

RSA:
Encryption: message $x \longmapsto$ ciphertext $x^e \bmod N = y$
Decryption: ciphertext $y \longmapsto$ plaintext $y^d \bmod N$
Encryption/decryption: $(x^e)^d \bmod N = x \bmod N$
Breaking RSA $\approx$ factoring N

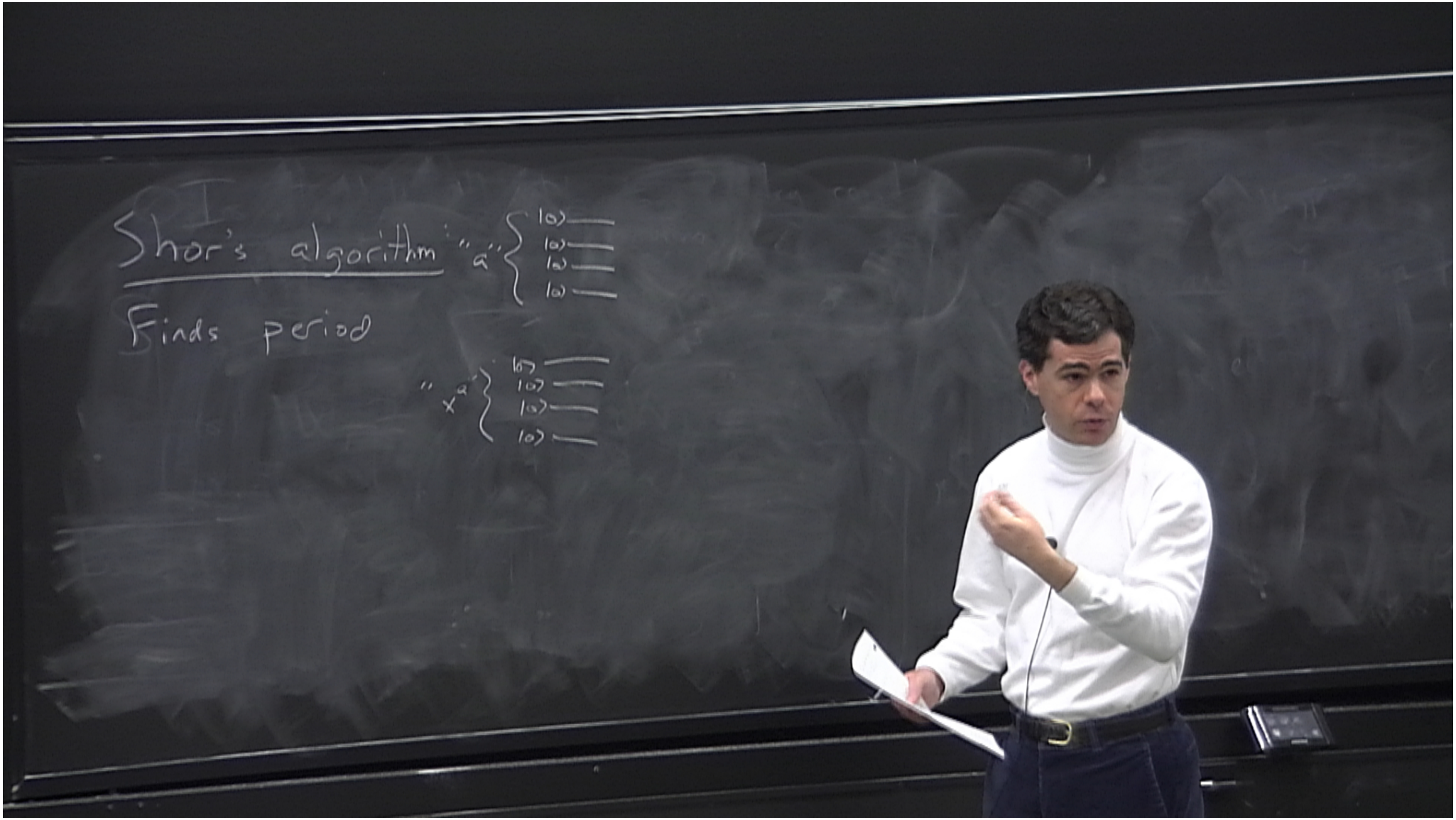

To factor, it is sufficient to find the period of $f(a) = x^a \bmod N$ (given x & N). I.e., find smallest $r > 0$ s.t. $x^r = 1 \bmod N$, r order of x.
We want to find some x w/

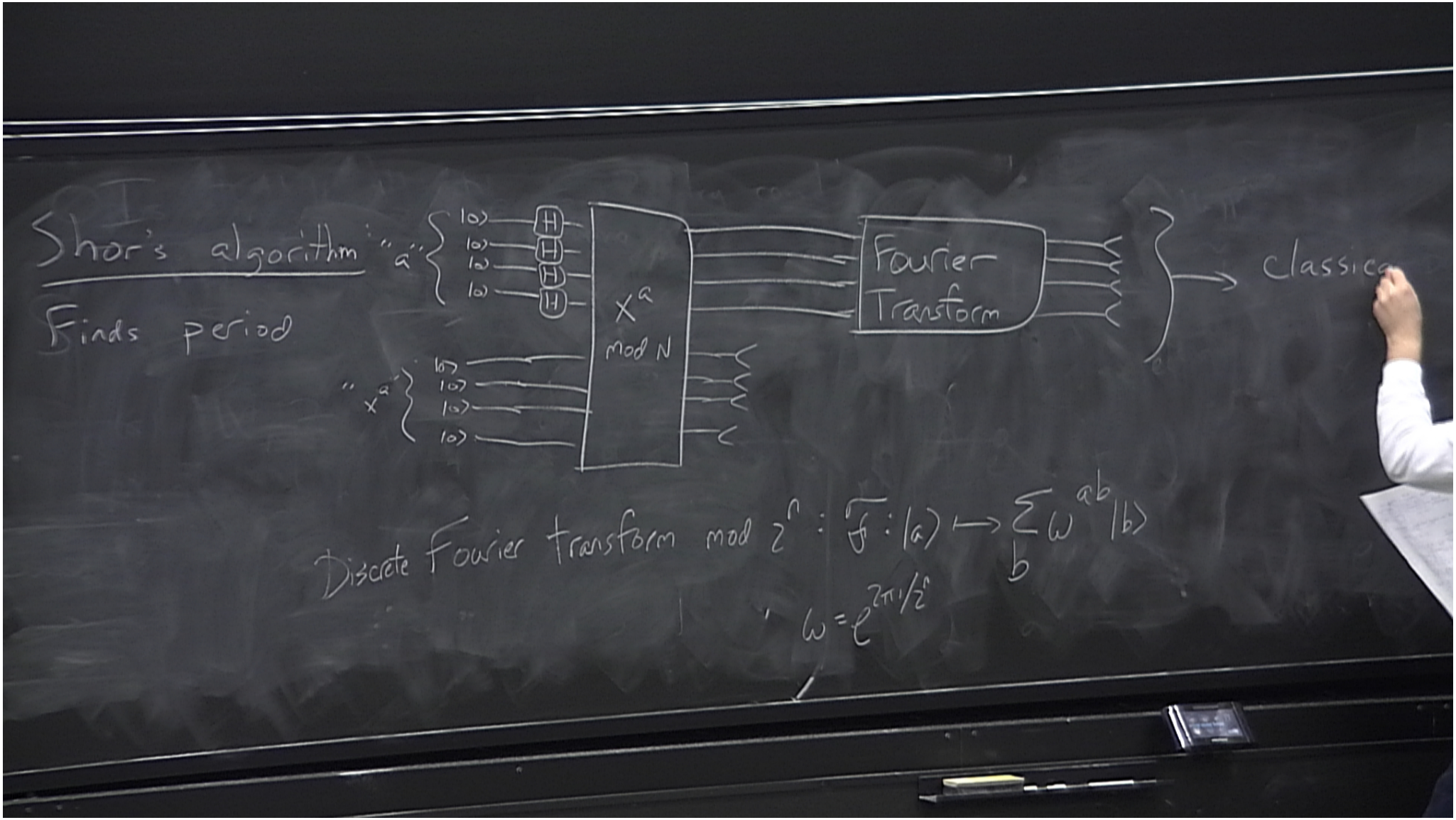

cient to
$f(a) = x^a \bmod N$
smallest $r > 0$
order of x.
We want to find some x w/ even order. If we pick x randomly, we find x w/ even order with constant probability.

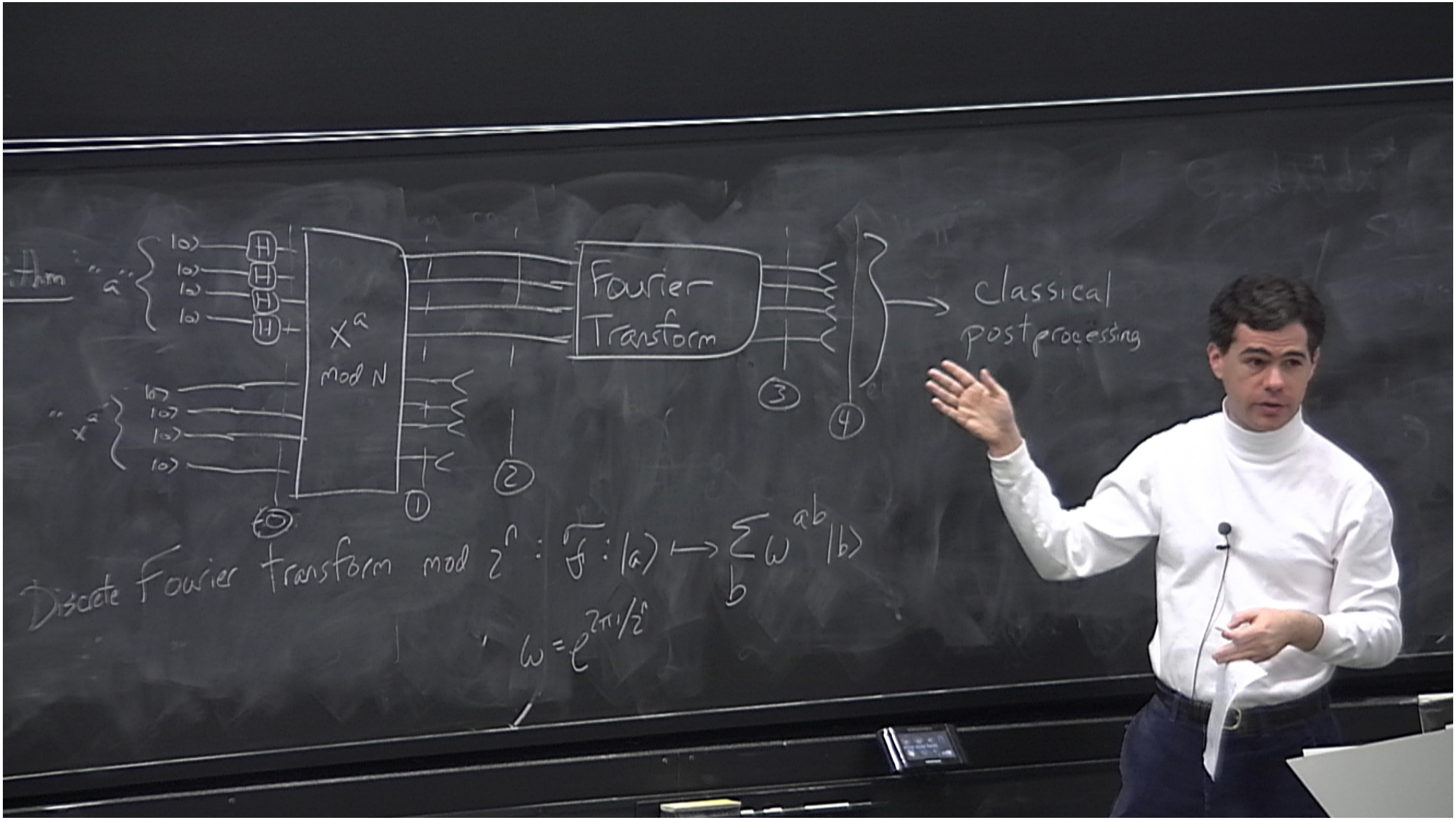

We want to find some x w/ even order. If we pick x randomly, we find x w/ even order with constant probability.
w/ constant prob, $x^{r/2} \neq -1 \bmod N$
$y = x^{r/2} \Rightarrow y^2 = 1 \bmod N$
$y^2 - 1 = (y+1)(y-1) = 0 \bmod N$
1) $N \mid y+1$
2) $N \mid y-1 \Rightarrow y = 1 \bmod N$ X
3) $p \mid y+1$ & $q \mid y-1$

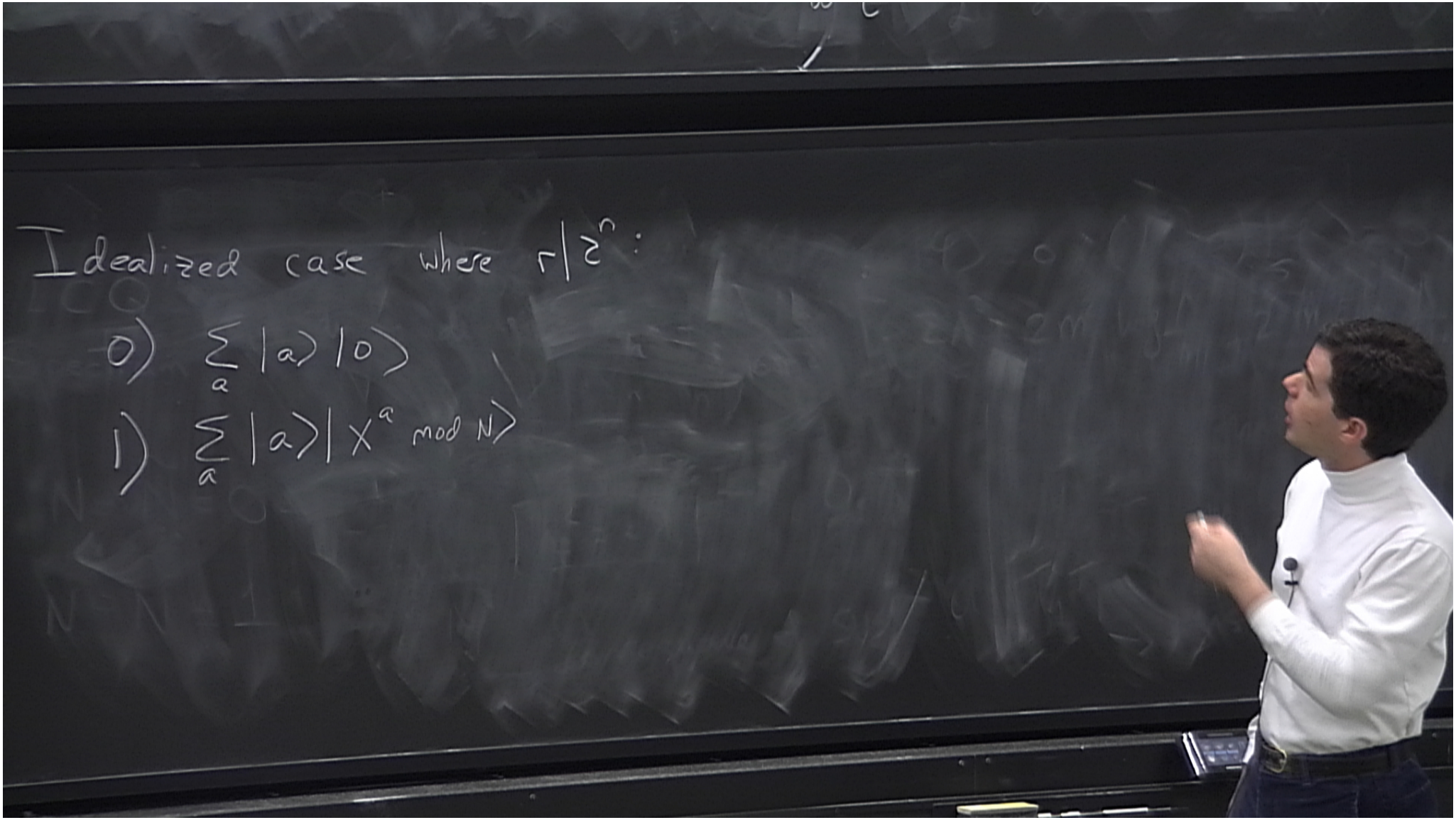

find some
der. If we
nly, we find
with constant
$x^{r/2} \neq -1 \bmod N$
$y = x^{r/2} \Rightarrow y^2 = 1 \bmod N$
$y^2 - 1 = (y+1)(y-1) = 0 \bmod N$
1) $N \mid y+1$ avoid
2) $N \mid y-1 \Rightarrow y = 1 \bmod N$ X
3) $p \mid y+1$ & $q \mid y-1$ $GCD(y+1, N) = p$
$q \nmid y+1$ & $p \nmid y-1$ $\Rightarrow$ Euclid's algorithm

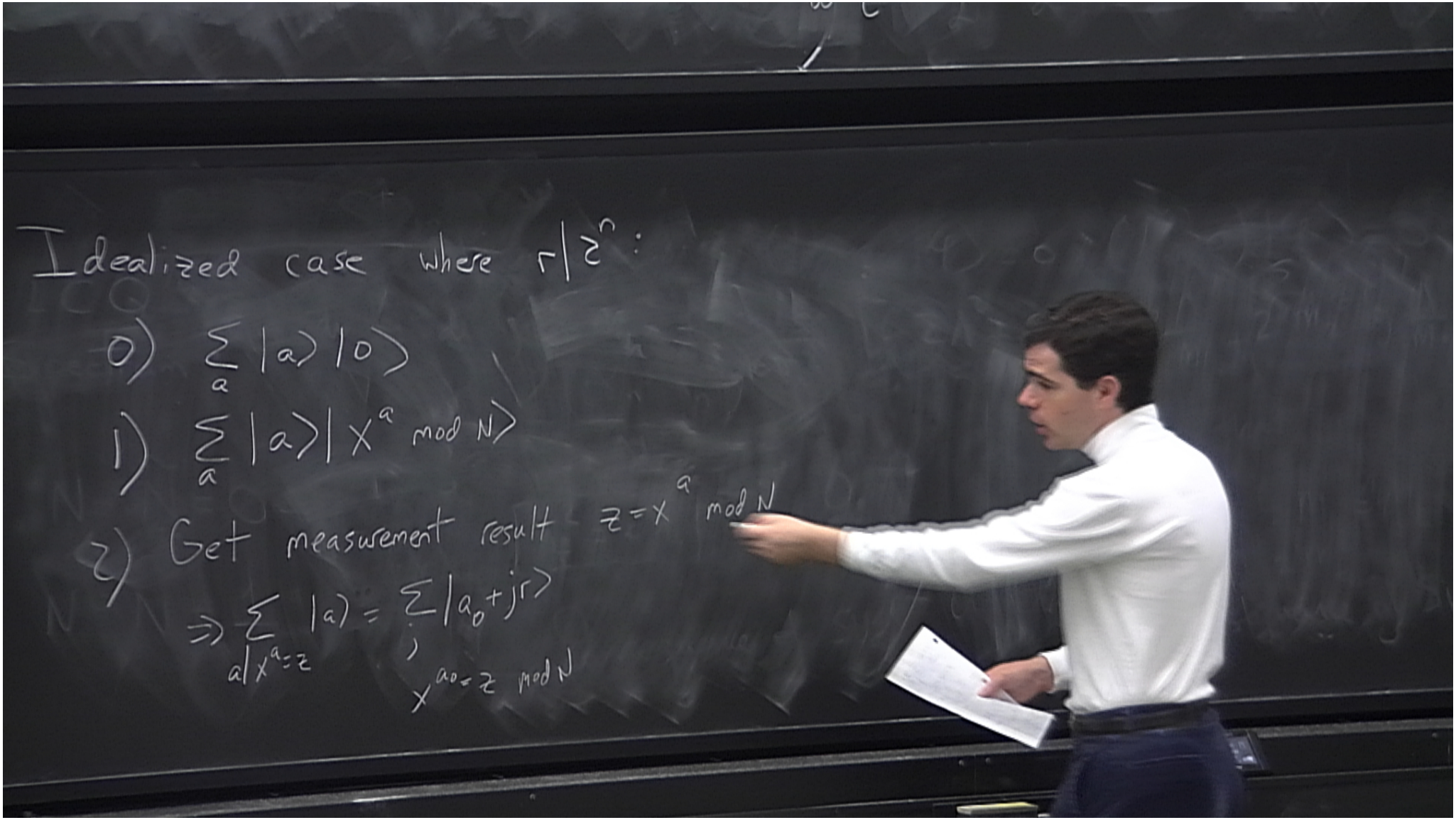

$y = x^{r/2} \Rightarrow y^2 = 1 \mod N$
$y^2 - 1 = (y+1)(y-1) = 0 \mod N$
1) $N \mid y+1$ avoid
2) $N \mid y-1 \Rightarrow y = 1 \mod N$ X
3) $p \mid y+1$ & $q \mid y-1$ $GCD(y+1, N) = p$
$q \nmid y+1$ & $p \nmid y-1$ $\Rightarrow$ Euclid's algorithm


Shor's algorithm
Finds period


Shor's algorithm
Finds period
"a"
|0⟩
|0⟩
|0⟩
|0⟩
H
H
H
H
x^a mod N
"x^a"
|0⟩
|0⟩
|0⟩
|0⟩
Fourier Transform
Classica
Discrete Fourier transform mod 2^n : $\mathcal{F}: |a\rangle \mapsto \sum_b \omega^{ab} |b\rangle$
$\omega = e^{2\pi i/2^n}$


Fourier Transform
Classical post processing
x^a mod N
Discrete Fourier transform mod 2^r : $\mathcal{F}: |a\rangle \mapsto \sum_b \omega^{ab} |b\rangle$
$\omega = e^{2\pi i/2^r}$


Idealized case where $r | 2^n$:
0) $\sum_a |a\rangle |0\rangle$
1) $\sum_a |a\rangle |x^a \bmod N\rangle$


Idealized case where $r | 2^n$:
0) $\sum_a |a\rangle |0\rangle$
1) $\sum_a |a\rangle |x^a \bmod N\rangle$
2) Get measurement result $z = x^a \bmod N$
$\Rightarrow \sum_{a | x^a = z} |a\rangle = \sum |a_0 + jr\rangle$
$x^{a_0} = z \bmod N$

Idealized case where $r \mid 2^n$:

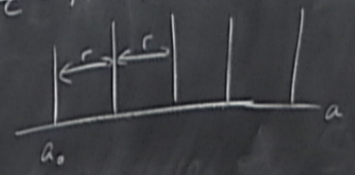
0) $\sum_a |a\rangle |0\rangle$

1) $\sum_a |a\rangle |x^a \bmod N\rangle$

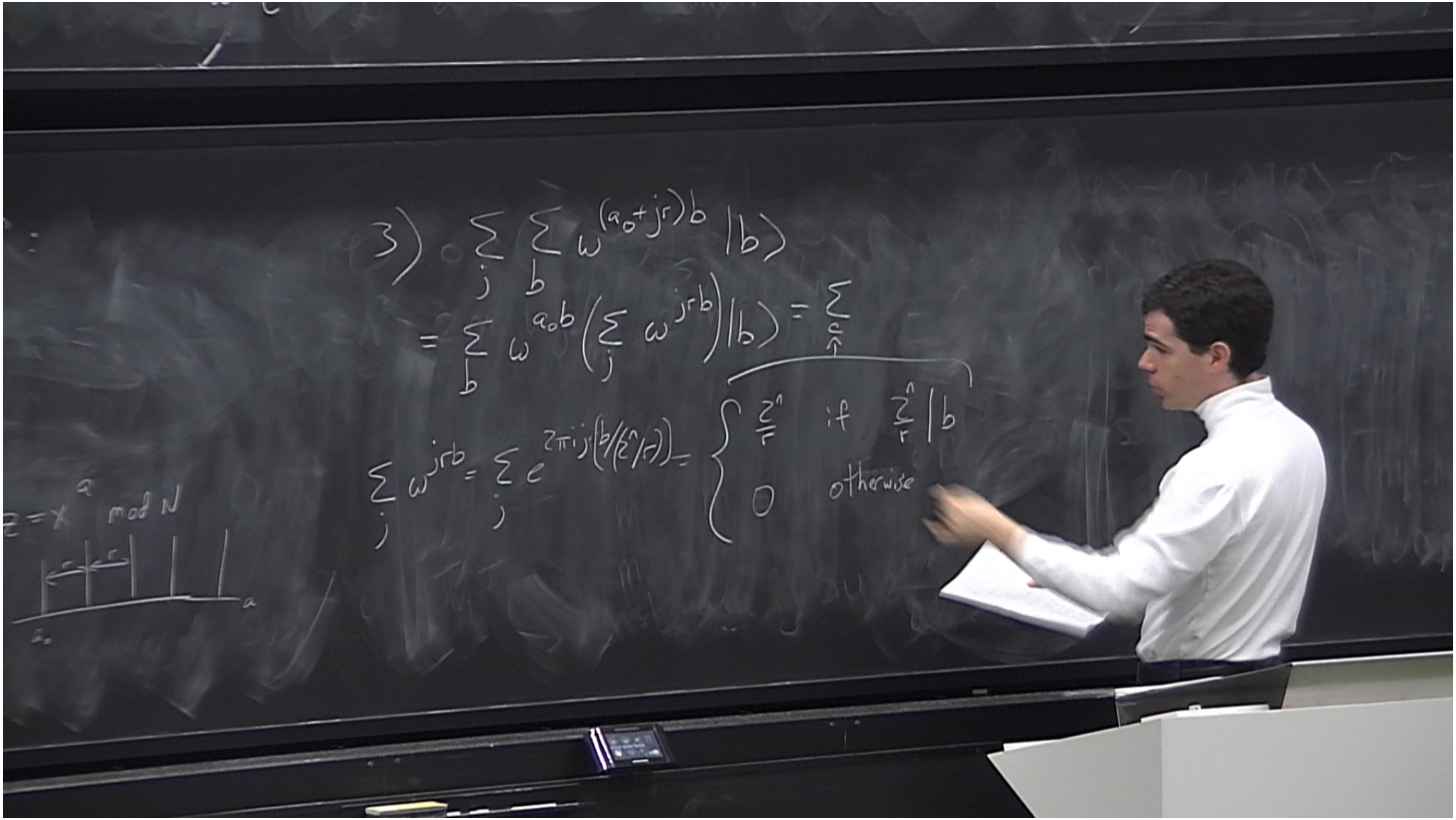
2) Get measurement result $z = x^{a_0} \bmod N$

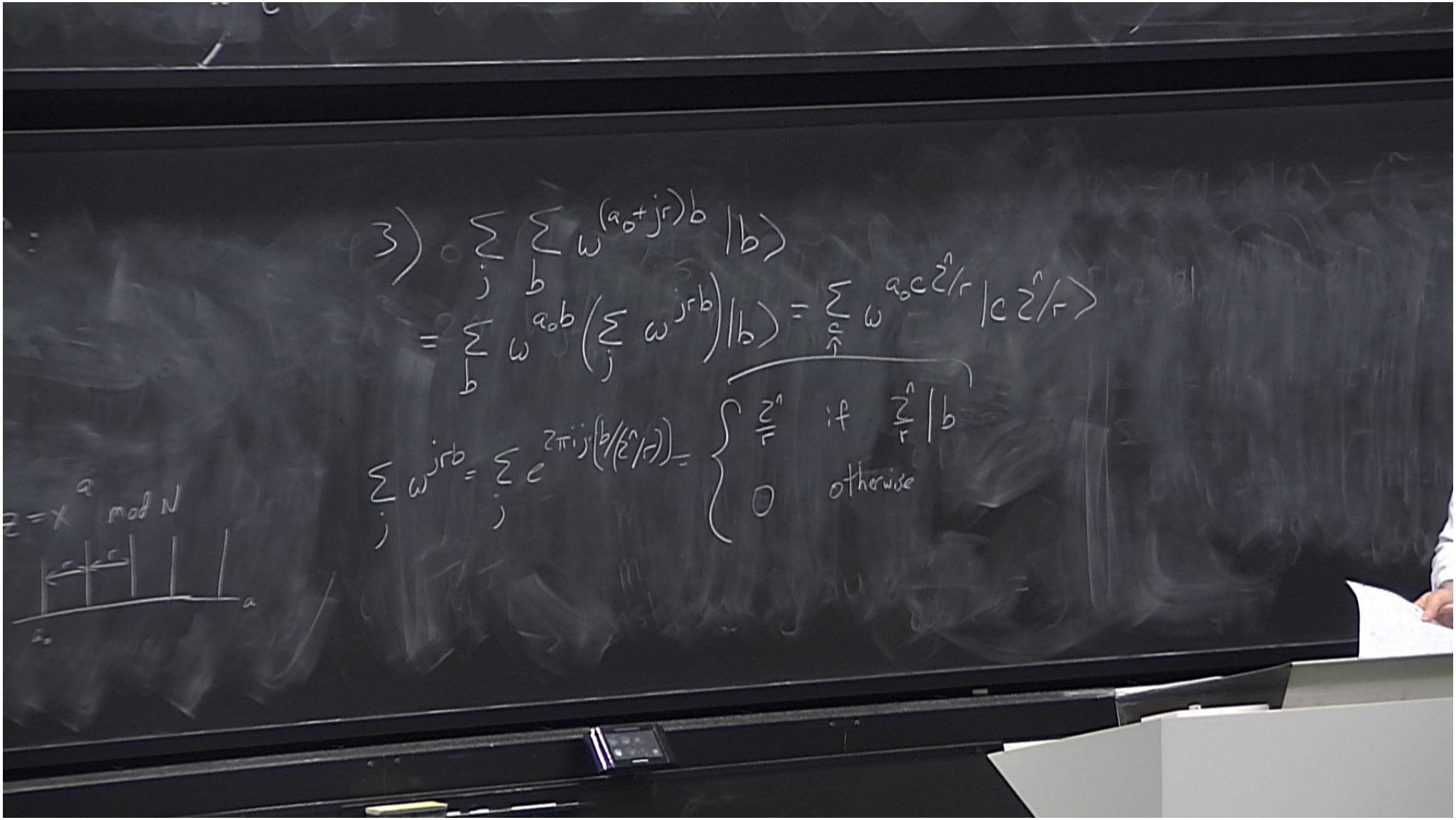
$\Rightarrow \sum_{a \mid x^a = z} |a\rangle = \sum_j |a_0 + jr\rangle$

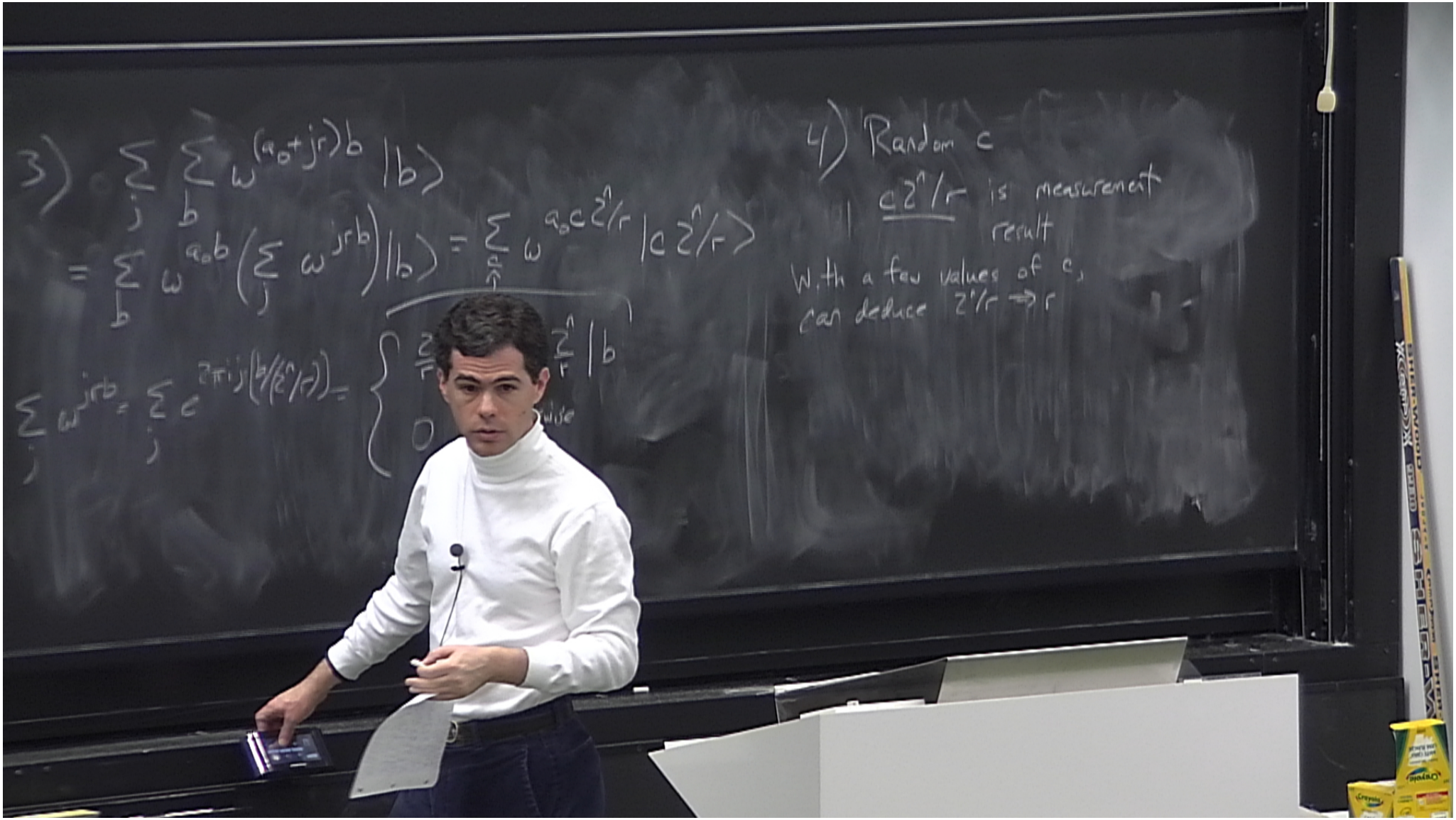
$x^{a_0} = z \bmod N$



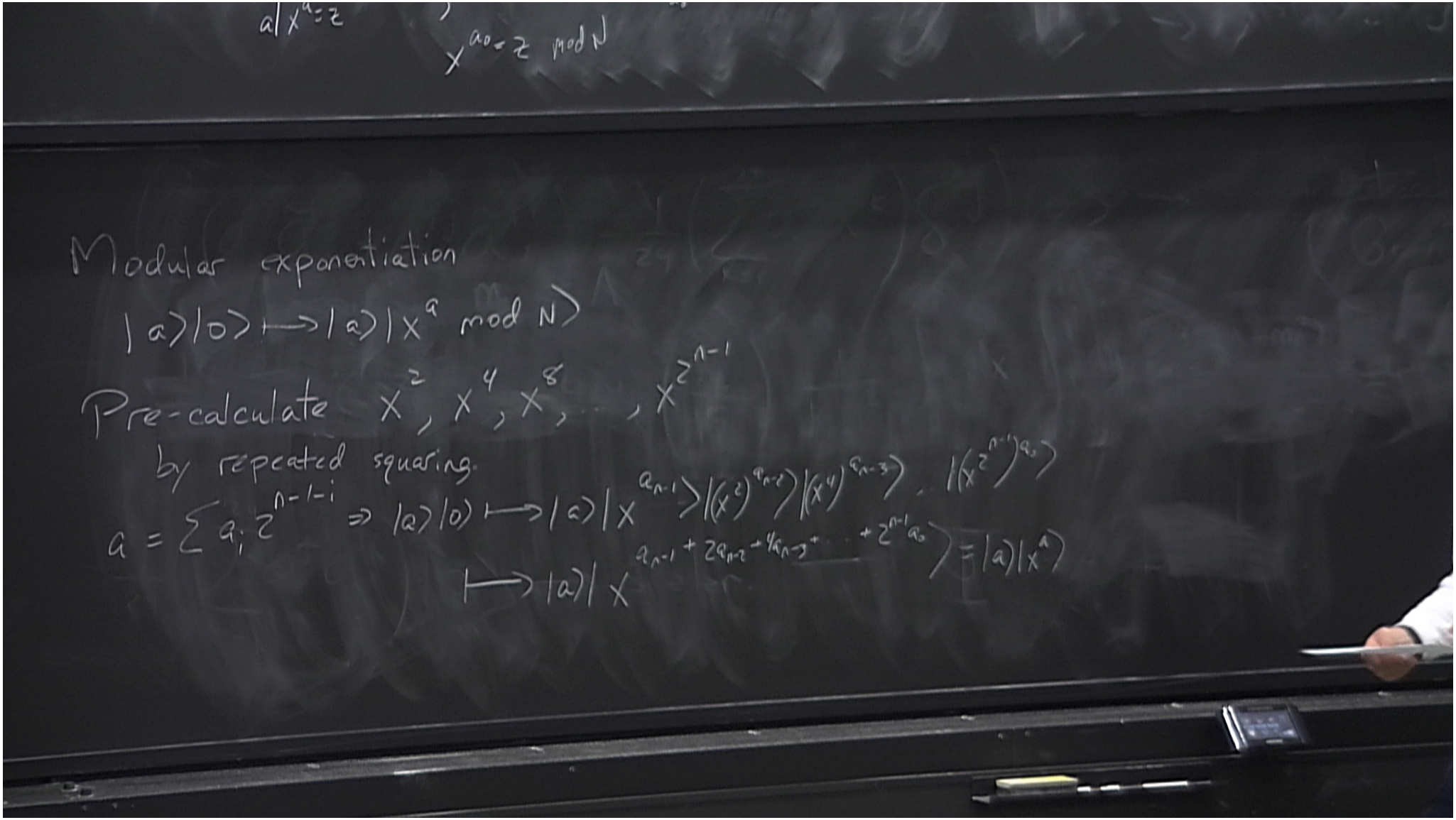
3) $\sum_j \sum_b \omega^{(a_0 + jr)b} |b\rangle$

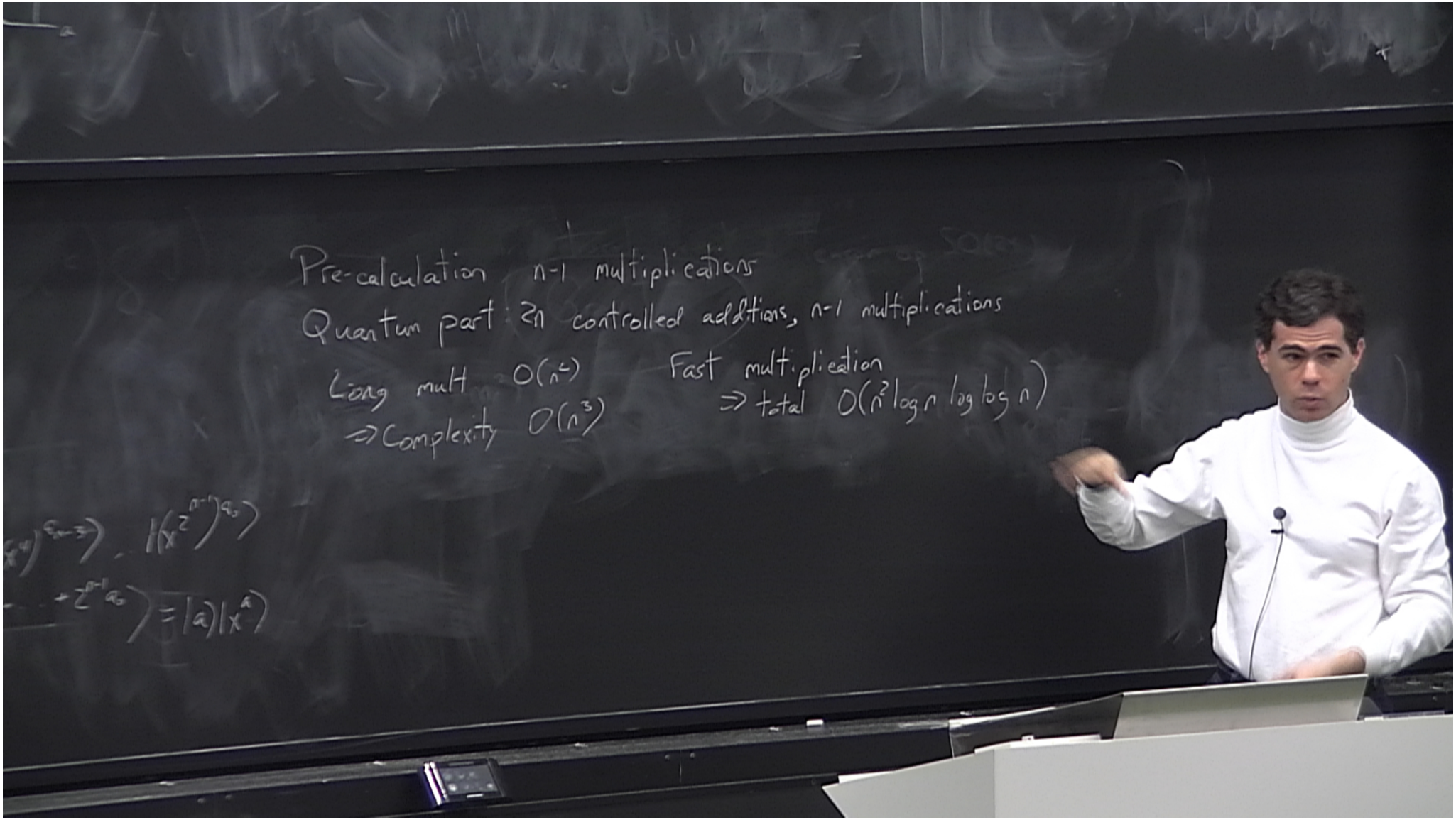

3)
$\sum_j \sum_b \omega^{(a_0+jr)b} |b\rangle$
$= \sum_b \omega^{a_0 b} \left(\sum_j \omega^{jrb}\right) |b\rangle =$
$\sum_j \omega^{jrb} = \sum_j e^{2\pi i j \left[b/(Q/r)\right]} =$
$\begin{cases} \frac{Q}{r} & \text{if } \frac{Q}{r} \mid b \\ 0 & \text{otherwise} \end{cases}$
$z = x^a \bmod N$
a_0
a

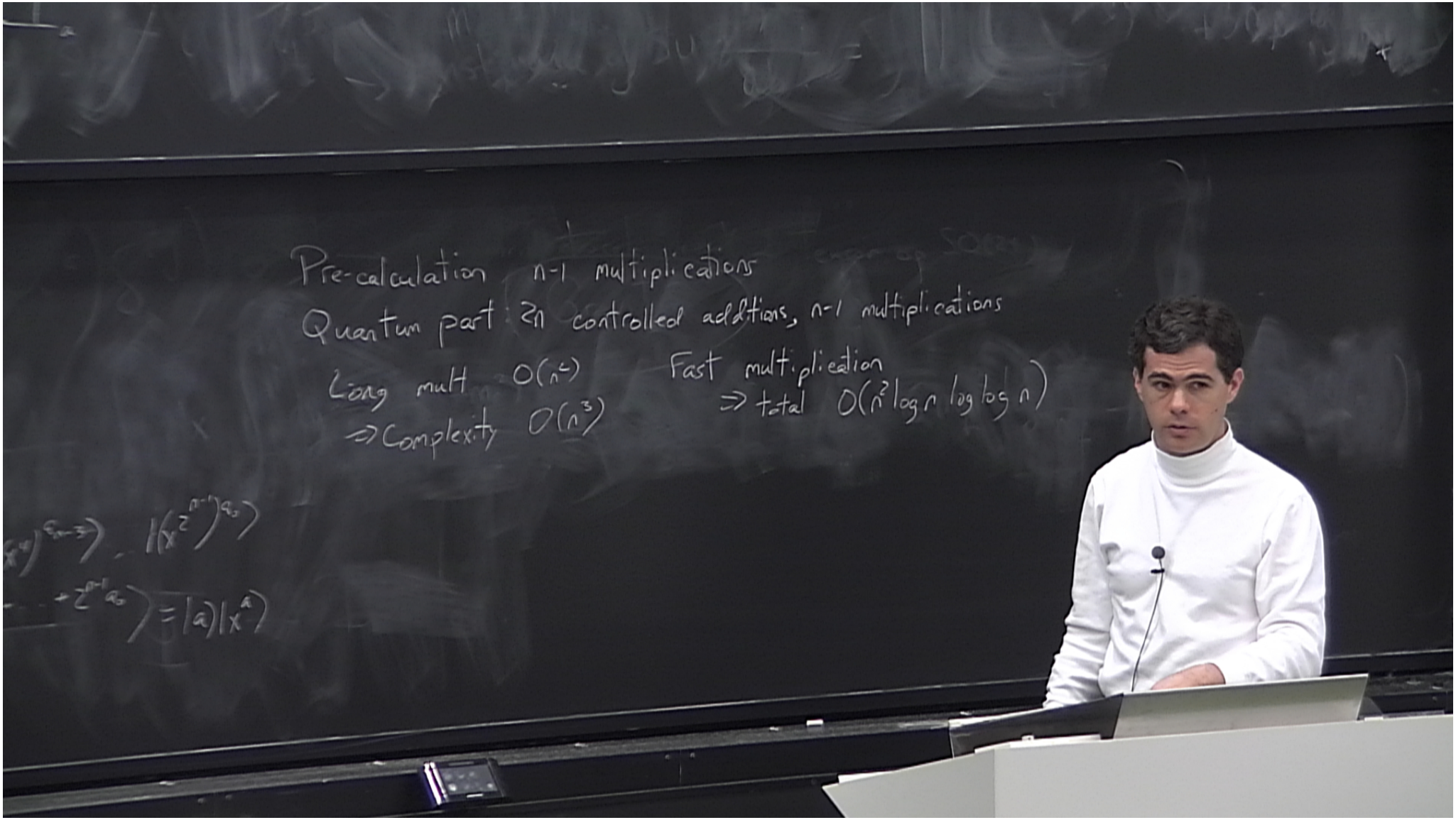

3) $\sum_j \sum_b \omega^{(a_0+jr)b} |b\rangle$
$= \sum_b \omega^{a_0 b} \left(\sum_j \omega^{jrb}\right) |b\rangle = \sum_c \omega^{a_0 c z/r} |c\, z/r\rangle$
$\sum_j \omega^{jrb} = \sum_j e^{2\pi i j \left[b/(z/r)\right]} = \begin{cases} \frac{z}{r} & \text{if } \frac{z}{r} \mid b \\ 0 & \text{otherwise} \end{cases}$
$z = x^a \bmod N$
a_0
a


4) Random c
c Z^/r is measurement result
With a few values of c, can deduce Z^/r → r


With a few values of c,
can deduce z'/r ⇒ r
otherwise


$a|x^a = z$
$x^{a_0} = z \bmod N$
Modular exponentiation
$|a\rangle|0\rangle \longmapsto |a\rangle|x^a \bmod N\rangle$
Pre-calculate $x^2, x^4, x^8, \ldots, x^{2^{n-1}}$
by repeated squaring.
$a = \sum a_i 2^{n-1-i} \Rightarrow |a\rangle|0\rangle \longmapsto |a\rangle|x^{a_{n-1}}\rangle|(x^2)^{a_{n-2}}\rangle|(x^4)^{a_{n-3}}\rangle \cdots |(x^{2^{n-1}})^{a_0}\rangle$
$\longmapsto |a\rangle|x^{a_{n-1} + 2a_{n-2} + 4a_{n-3} + \cdots + 2^{n-1}a_0}\rangle = |a\rangle|x^a\rangle$


Pre-calculation n-1 multiplications
Quantum part: 2n controlled additions, n-1 multiplications
Long mult $O(n^2)$
⇒ Complexity $O(n^3)$
Fast multiplication
⇒ total $O(n^2 \log n \log\log n)$


Pre-calculation n-1 multiplications
Quantum part: 2n controlled additions, n-1 multiplications
Long mult $O(n^2)$
⇒ Complexity $O(n^3)$
Fast multiplication
⇒ total $O(n^2 \log n \log\log n)$