Title: Quantum Information - Review (PHYS 635) - Lecture 7

Date: Feb 02, 2010 09:00 AM

URL: http://pirsa.org/10020039

Abstract: Week 1: Basic topics (Qubits, quantum gates, quantum circuits, density matrices, quantum operations, entropy, entanglement) Week 2: Algorithms and complexity (Languages, complexity classes, oracles, RSA, Deutsch-Jozsa algorithm, Shor's algorithm, Grover's algorithm) Week 3: Information theory and implementations (Overview of implementations, quantum error correction, quantum cryptography, quantum information theory)

Many languages can be decided, but many languages <u>cannot</u> be decided on a Turing machine.

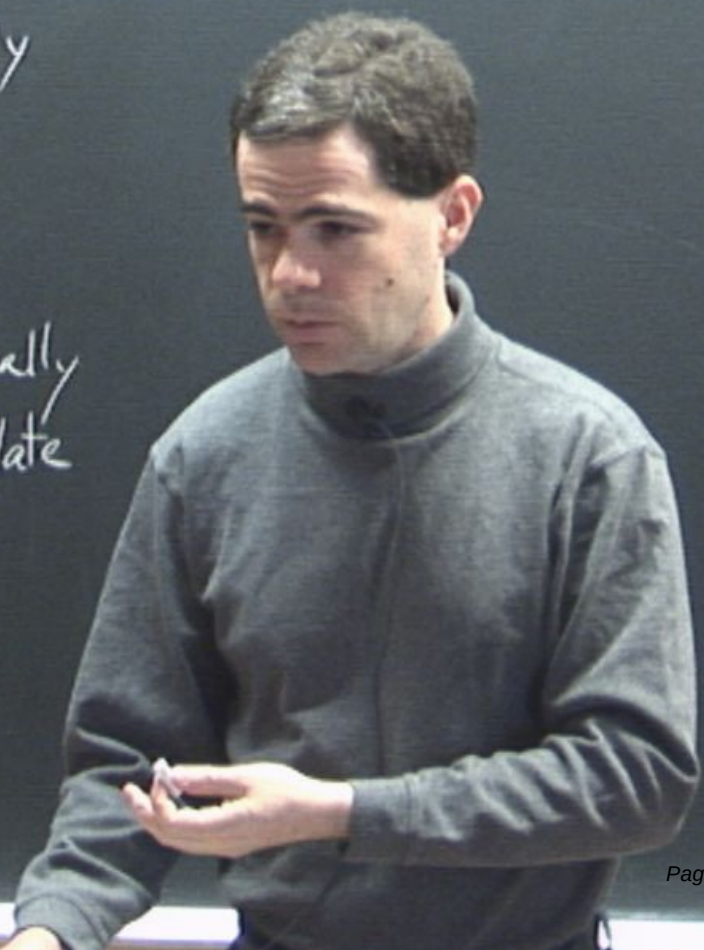
Many languages can be decided, but many languages cannot be decided on a Turing machine.

Church-Turing thesis: All physically realistic models of computation have the same set of decidable languages.

Many languages can be decided, but many languages cannot be decided on a Turing machine.

Church-Turing thesis: All physically realistic models of computation have the same set of decidable languages.

Strong Church-Turing thesis: All physically realistic models of computation can simulate each other using polynomial resources.



Many languages can be decided, but many languages <u>cannot</u> be decided on a Turing machine.

<u>Church-Turing thesis</u>: All physically realistic models of computation have the same set of decidable languages.

<u>Strong Church-Turing thesis</u>: All physically realistic models of computation can simulate each other using polynomial resources. In particular, these models have the same class of poly-time computable functions.

Many languages can be decided, but many languages <u>cannot</u> be decided on a Turing machine.

<u>Church-Turing thesis</u>: All physically realistic models of computation have the same set of decidable languages.

<u>Strong Church-Turing thesis</u>: All physically realistic models of computation can simulate each other using polynomial resources. In particular, these models have the same class of poly-time computable functions.

→ Probably false, because quantum computers probably cannot be efficiently simulated by classical computers.

<u>Def</u>:

Many languages can be decided, but many languages <u>cannot</u> be decided on a Turing machine.

<u>Church-Turing thesis</u>: All physically realistic models of computation have the same set of decidable languages.

<u>Strong Church-Turing thesis</u>: All physically realistic models of computation can simulate each other using polynomial resources. In particular, these models have the same class of poly-time computable functions.

→ Probably false, because quantum computers probably cannot be efficiently simulated by classical computers.

<u>Def.</u>: BQP ("bounded quantum polynomial") is class of languages L s.t. $\exists$ poly-time quantum algorithm $A(x)$ s.t. a) if $x \in L$, $A(x) = \text{yes}$ w/ prob. $\geq \frac{2}{3}$, b) if $x \notin L$, $A(x) = \text{no}$ w/ prob. $\geq \frac{2}{3}$

Many languages can be decided, but many languages cannot be decided on a Turing machine.

Church-Turing thesis: All physically realistic models of computation have the same set of decidable languages.

Strong Church-Turing thesis: All physically realistic models of computation can simulate each other using polynomial resources. In particular, these models have the same class of poly-time computable functions.

Probably false, because quantum computers probably cannot be efficiently simulated by classical computers.

Def: BQP ("bounded quantum polynomial") is class of languages L s.t. $\exists$ poly-time quantum algorithm $A(x)$ s.t. a) if $x \in L$, $A(x) = \text{yes}$ w/ prob. $\geq \frac{2}{3}$, b) if $x \notin L$, $A(x) = \text{no}$ w/ prob. $\geq \frac{2}{3}$

Can repeat to amplify prob.

Many languages can be decided, but many languages <u>cannot</u> be decided on a Turing machine.

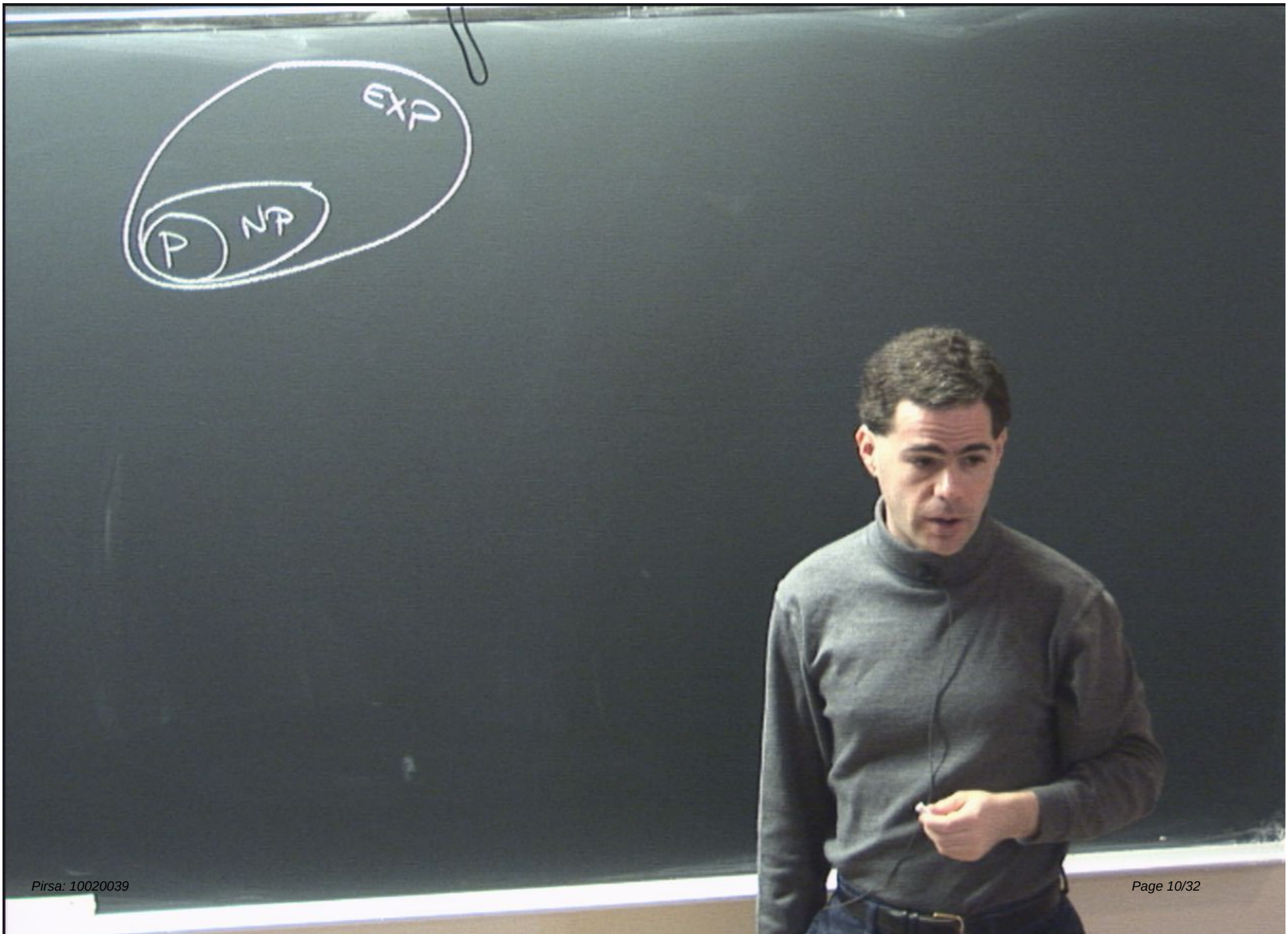
<u>Church-Turing thesis</u>: All physically realistic models of computation have the same set of decidable languages.

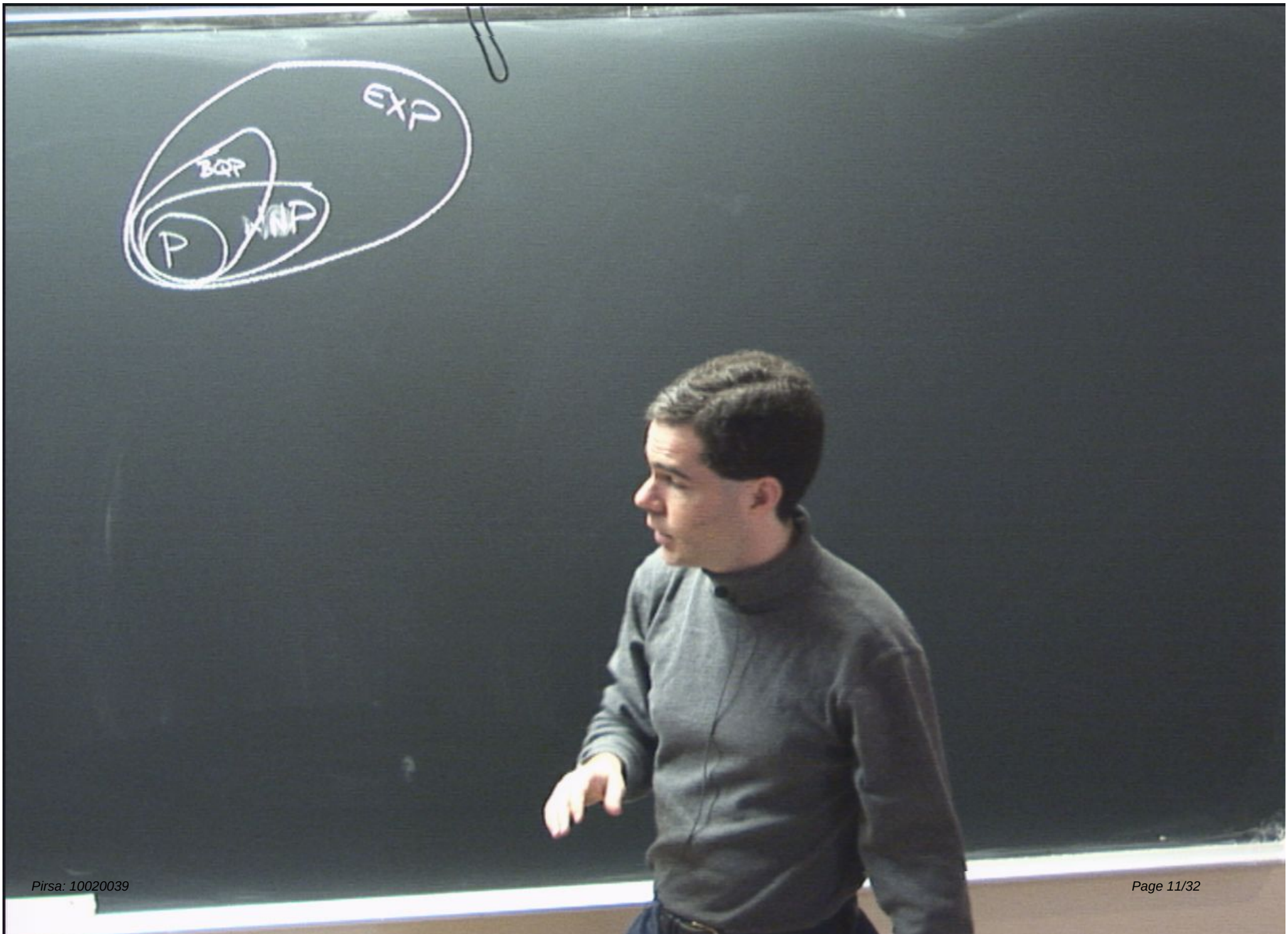
<u>Strong Church-Turing thesis</u>: All physically realistic models of computation can simulate each other using polynomial resources. In particular, these models have the same class of poly-time computable functions.

→ Probably false, because quantum computers probably cannot be efficiently simulated by classical computers.

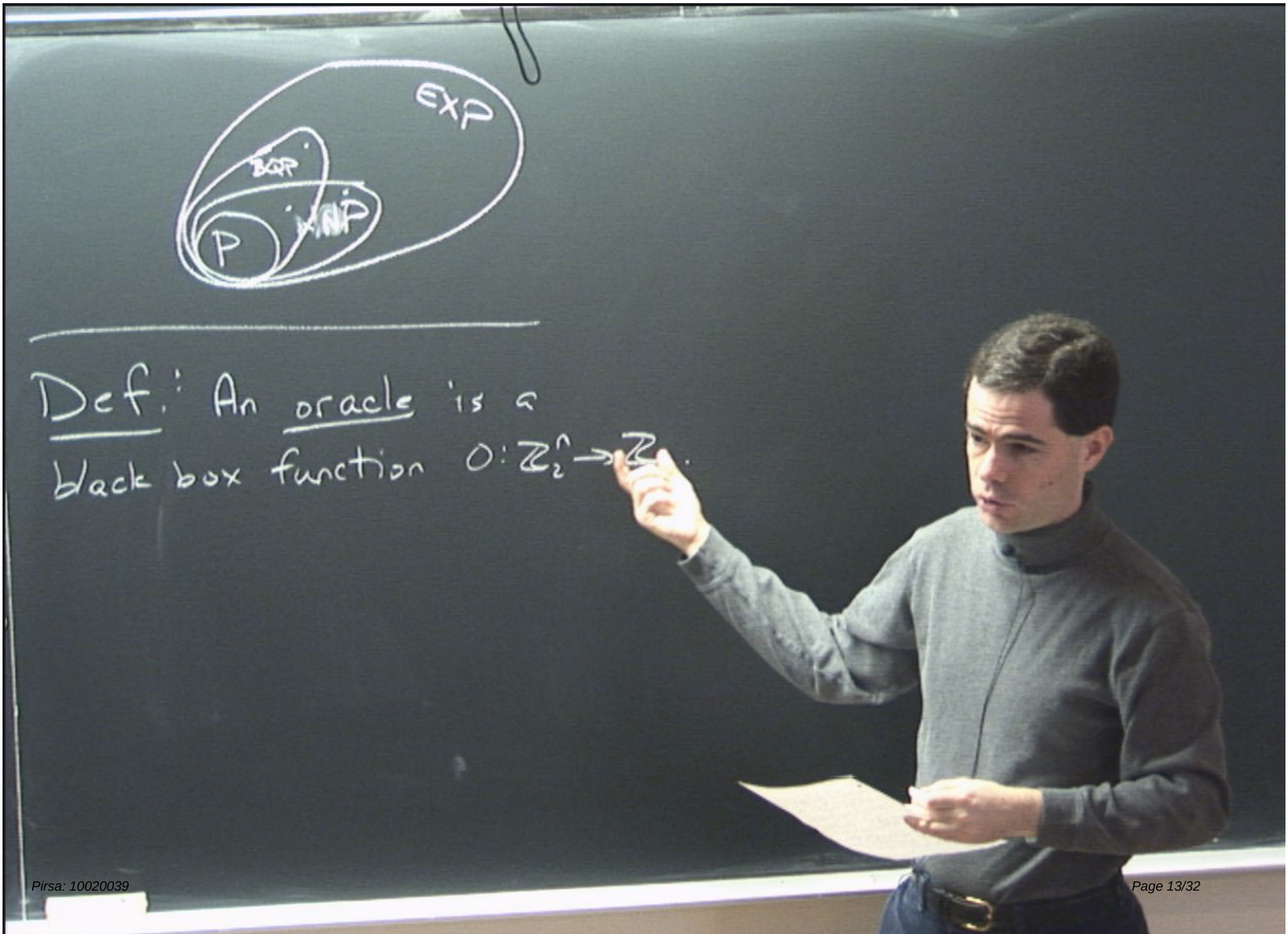
<u>Def.</u>: BQP ("bounded quantum polynomial") is class of languages L s.t. $\exists$ poly-time quantum algorithm $A(x)$ s.t. a) if $x \in L$, $A(x) = \text{yes}$ w/ prob. $\geq \frac{2}{3}$, b) if $x \notin L$, $A(x) = \text{no}$ w/ prob. $\geq \frac{2}{3}$

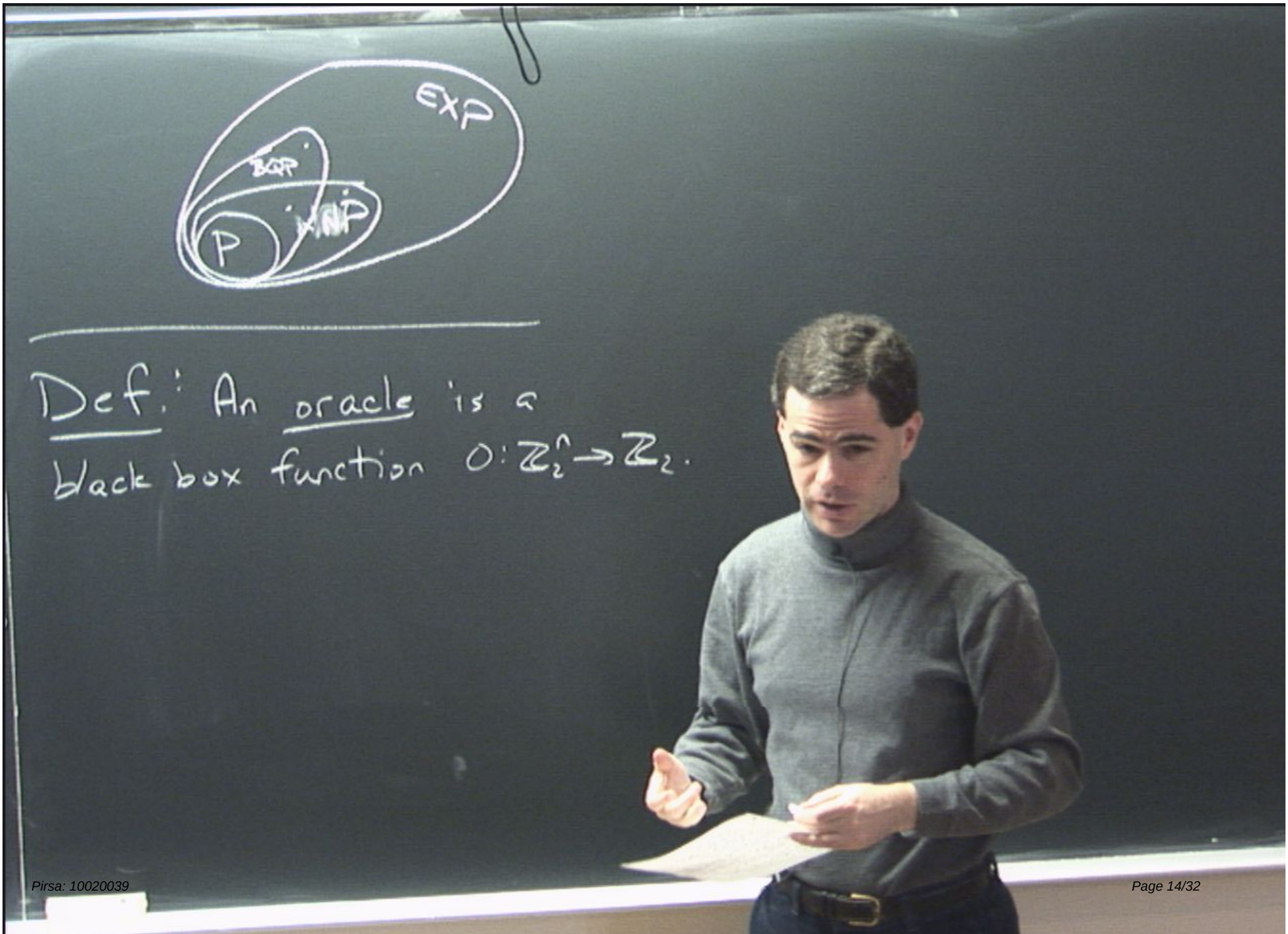
Can repeat to amplify prob.

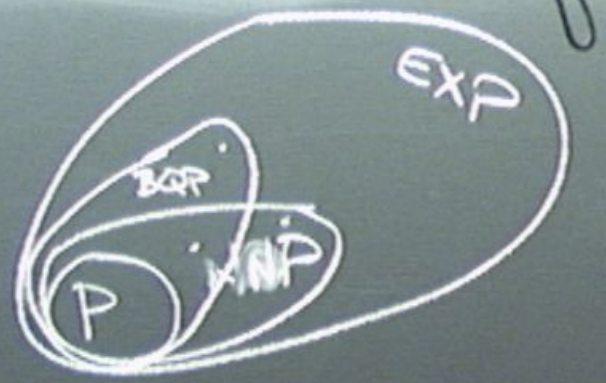

EXP
NP
P


EXP
BQP
NP
P


EXP
BQP
NP
P

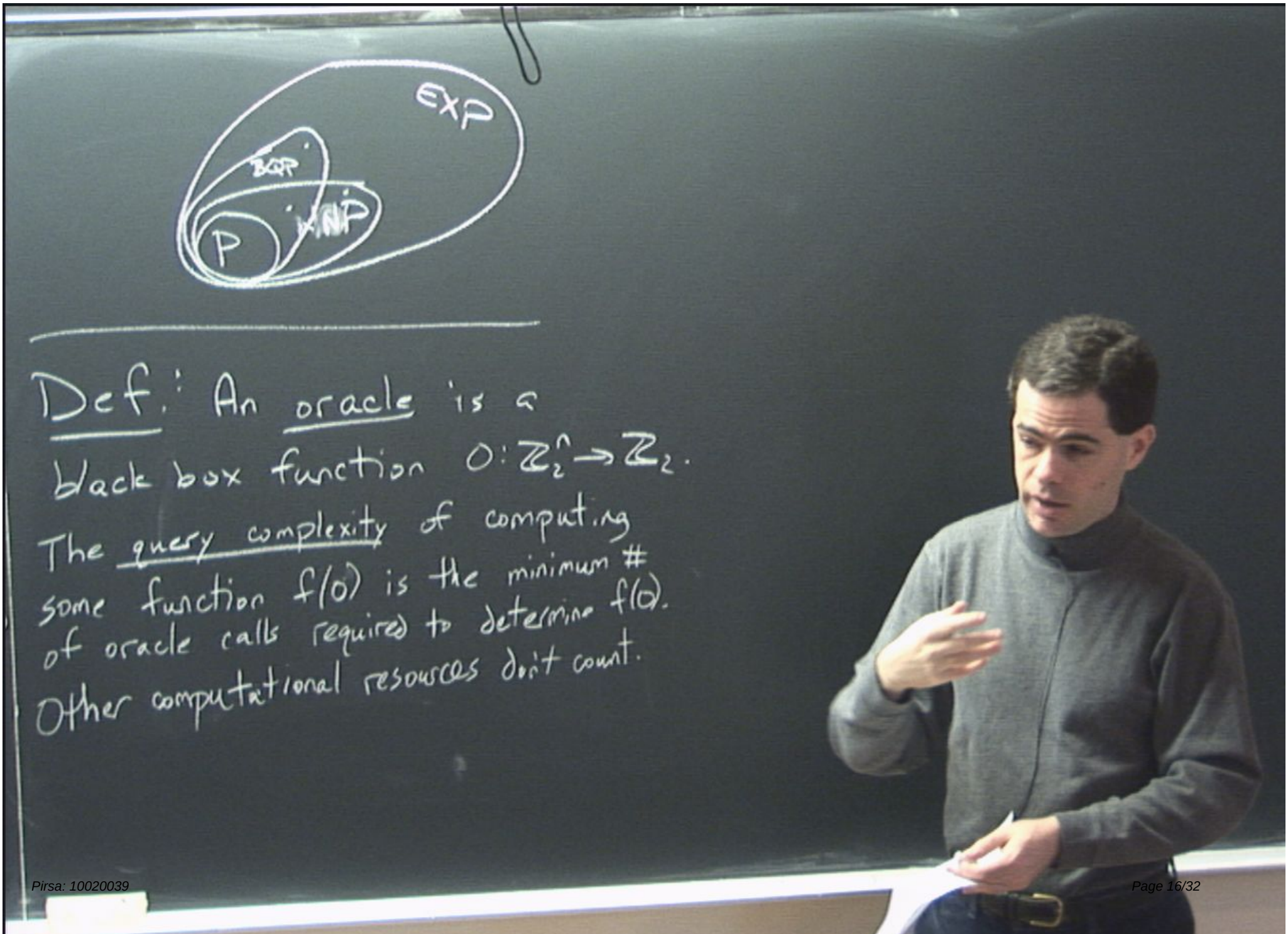

EXP
BQP
NP
P
Def.: An oracle is a
black box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.

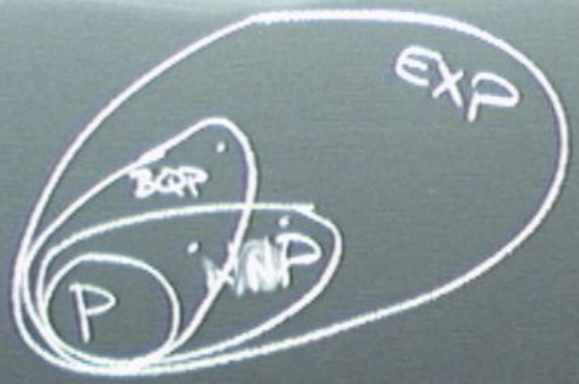

EXP
BQP
NP
P
Def.: An oracle is a
black box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.



Def.: An <u>oracle</u> is a black box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.

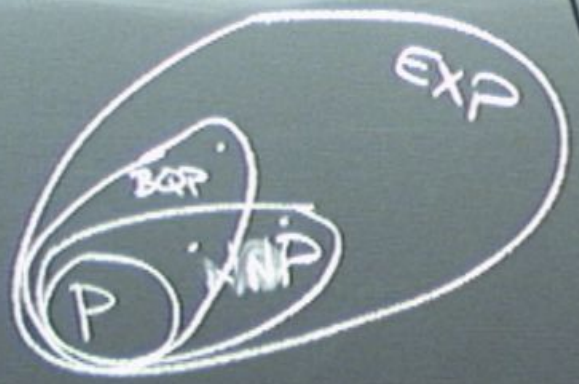
The <u>query complexity</u> of computing some function $f(O)$ is the minimum # of oracle calls required to determine $f(O)$.


EXP
BQP
NP
P
Def.: An oracle is a
black box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.
The query complexity of computing
some function $f(O)$ is the minimum #
of oracle calls required to determine $f(O)$.
Other computational resources don't count.



: An <u>oracle</u> is a
box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.
<u>uery complexity</u> of computing
function $f(O)$ is the minimum #
acle calls required to determine $f(O)$.
computational resources don't count.

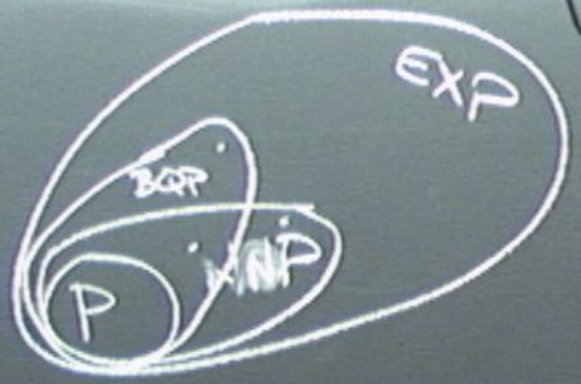
Suppose $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$ is
either a <u>constant</u> function $(O(x) = O(y)\ \forall x,y)$
or a <u>balanced</u> function $\left(|\{x \mid O(x)=0\}| = |\{x \mid O(x)=1\}|\right)$



: An <u>oracle</u> is a
box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.
<u>uery complexity</u> of computing
function $f(O)$ is the minimum #
acle calls required to determine $f(O)$.
computational resources don't count.

Suppose $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$ is
either a <u>constant</u> function $(O(x) = O(y)\ \forall x,y)$
or a <u>balanced</u> function $(|\{x \mid O(x)=0\}| = |\{x \mid O(x)=1\}|)$

Given an oracle O from this
family, determine whether it
is constant or balanced.

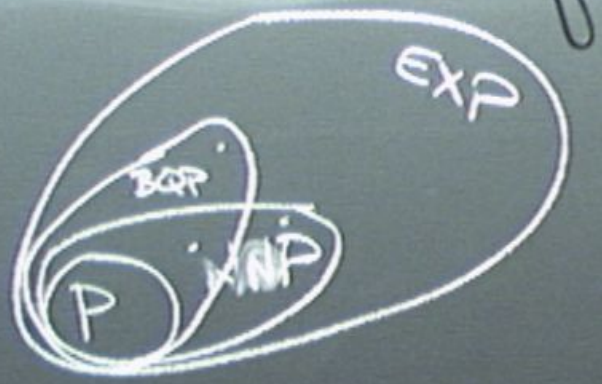


: An <u>oracle</u> is a
box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.
<u>uery complexity</u> of computing
function $f(O)$ is the minimum #
acle calls required to determine $f(O)$.
computational resources don't count.

Suppose $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$ is
either a <u>constant</u> function $(O(x) = O(y)\ \forall x,y)$
or a <u>balanced</u> function $(|\{x \mid O(x)=0\}| = |\{x \mid O(x)=1\}|)$

Given an oracle O from this
family, determine whether it
is constant or balanced.

To be completely sure, which one,
$2^{n-1}+1$ queries are needed.

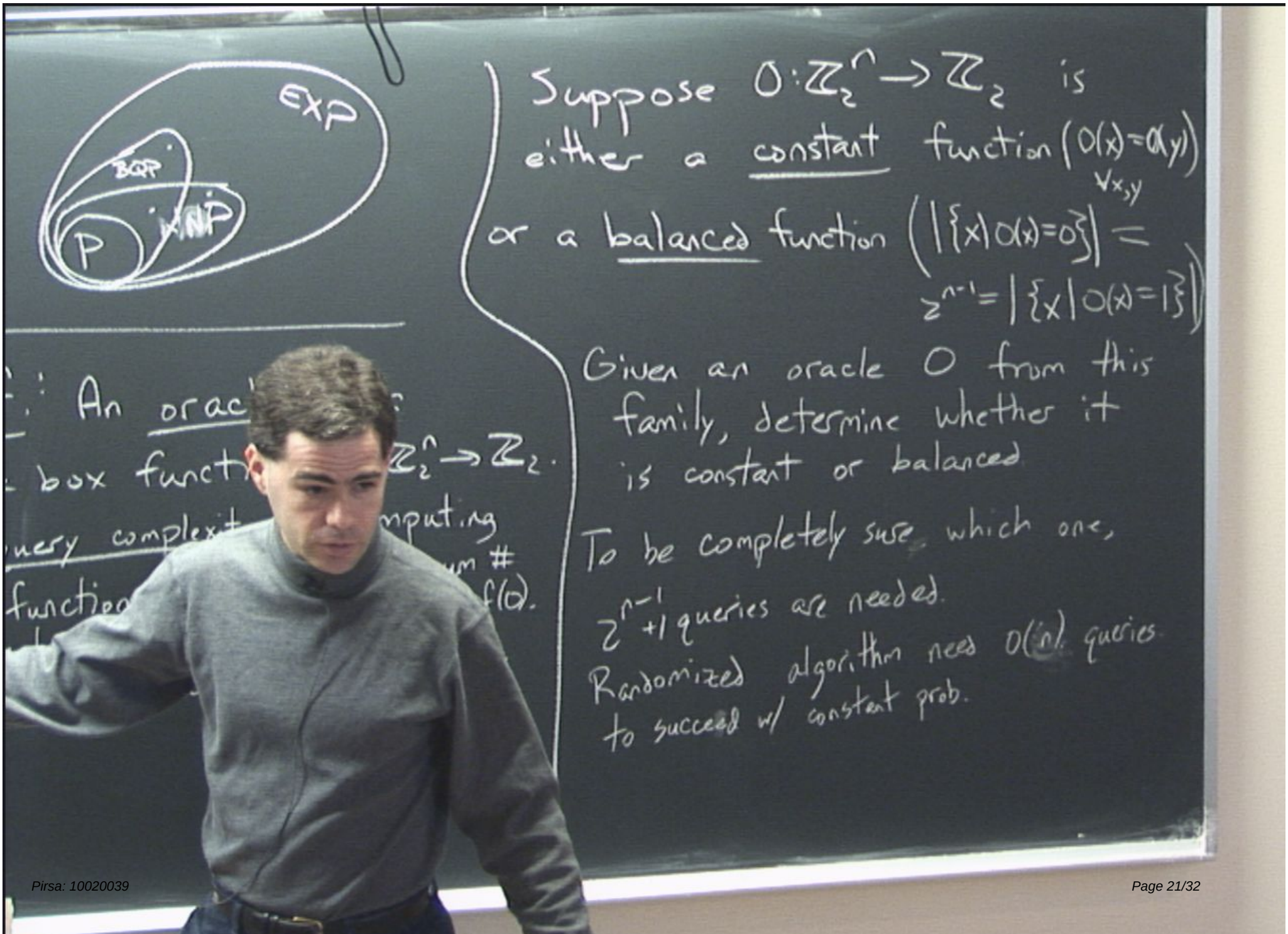


: An <u>oracle</u> is a
box function $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$.
<u>uery complexity</u> of computing
function $f(O)$ is the minimum #
acle calls required to determine $f(O)$.
computational resources don't count.

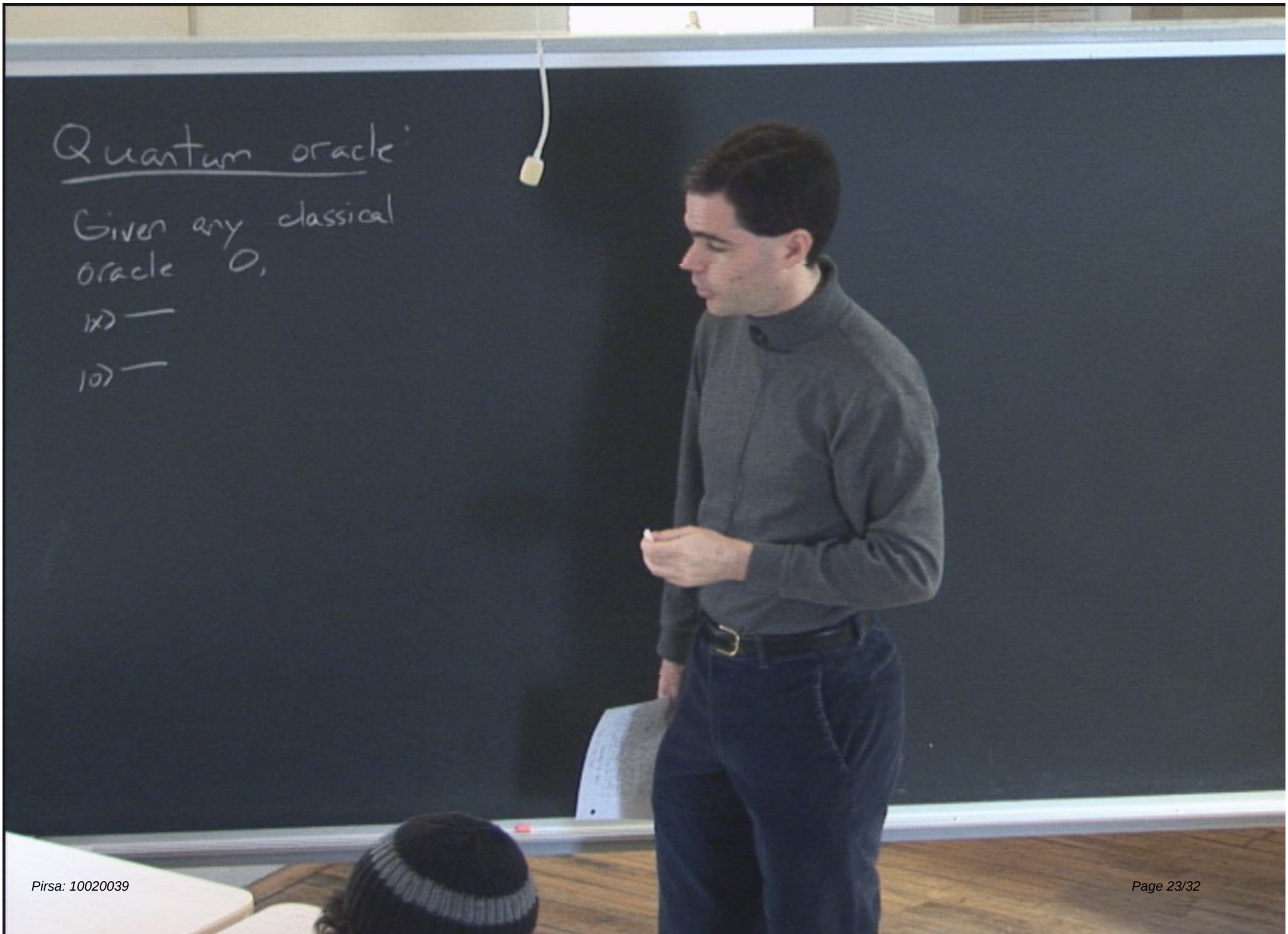
Suppose $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$ is
either a <u>constant</u> function $(O(x) = O(y)\ \forall x,y)$
or a <u>balanced</u> function $\left(|\{x \mid O(x)=0\}| = 2^{n-1} = |\{x \mid O(x)=1\}|\right)$

Given an oracle O from this family, determine whether it is constant or balanced

To be completely sure, which one, $2^{n-1}+1$ queries are needed.
Randomized algorithm need $O(n)$

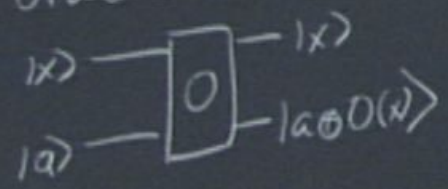

EXP
BQP
NP
P
Suppose $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$ is
either a constant function ($O(x)=O(y)$ $\forall x,y$)
or a balanced function ($|\{x \mid O(x)=0\}| = 2^{n-1} = |\{x \mid O(x)=1\}|$)
Given an oracle O from this family, determine whether it is constant or balanced
To be completely sure, which one,
$2^{n-1}+1$ queries are needed.
Randomized algorithm needs $O(n)$ queries
to succeed w/ constant prob.


EXP
BQP
NP
P
Suppose $O: \mathbb{Z}_2^n \to \mathbb{Z}_2$ is
either a constant function ($O(x) = O(y)$ $\forall x, y$)
or a balanced function ($|\{x \mid O(x)=0\}| = 2^{n-1} = |\{x \mid O(x)=1\}|$)
Given an oracle O from this family, determine whether it is constant or balanced
To be completely sure, which one,
$2^{n-1}+1$ queries are needed.
Randomized algorithm need $O(n)$ queries
to succeed w/ constant prob.


Quantum oracle:
Given any classical
oracle O,
|x⟩
|0⟩

Quantum oracle:

Given any classical oracle O,



Here's a quantum algorithm for the constant/balanced oracle problem

Deutsch-Jozsa algorithm

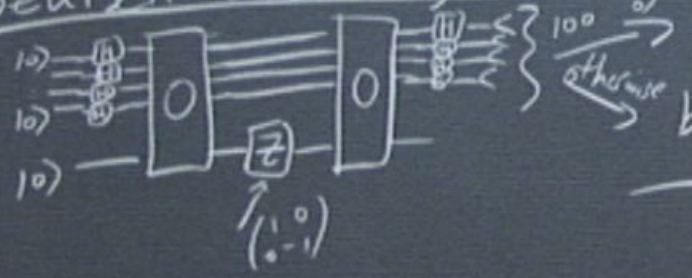
oracle:

lassical

$O(x)\rangle$

Here's a quantum algorithm for the constant/balanced oracle problem

Deutsch-Jozsa algorithm



Constant:

$|0\ldots0\rangle|0\rangle \rightarrow$

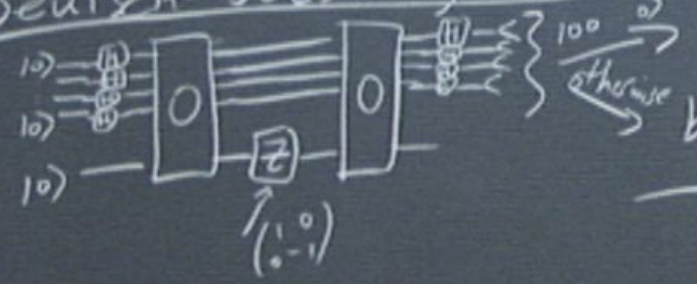
oracle:

lassical

$O(x)\rangle$

Here's a quantum algorithm for the constant/balanced oracle problem.

Deutsch-Jozsa algorithm



Constant:

$|0\rangle \quad 0\rangle|0\rangle \to \left(\sum_x |x\rangle\right) \otimes |0\rangle$

$\to \left(\sum_x |x\rangle \otimes |O(x)\rangle\right) = \left(\sum_x |x\rangle\right) \otimes |O(x)\rangle$

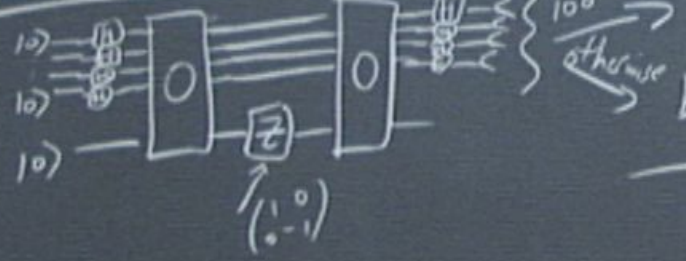
oracle:

lassical

$O(x)\rangle$

Here's a quantum algorithm for the constant/balanced oracle problem.

Deutsch-Jozsa algorithm



Constant:

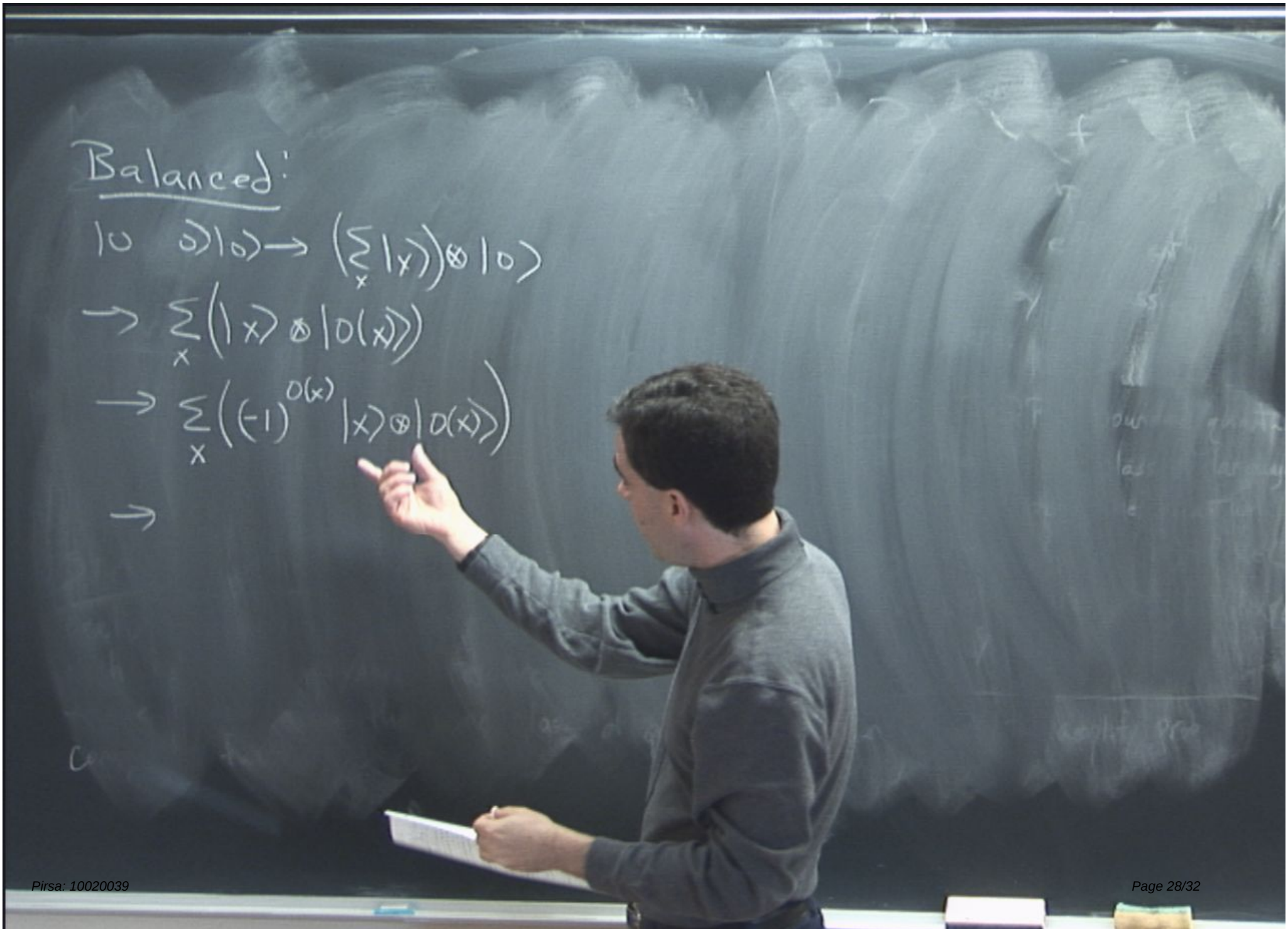
$$|0\ldots0\rangle|0\rangle \to \left(\sum_x |x\rangle\right)\otimes|0\rangle$$

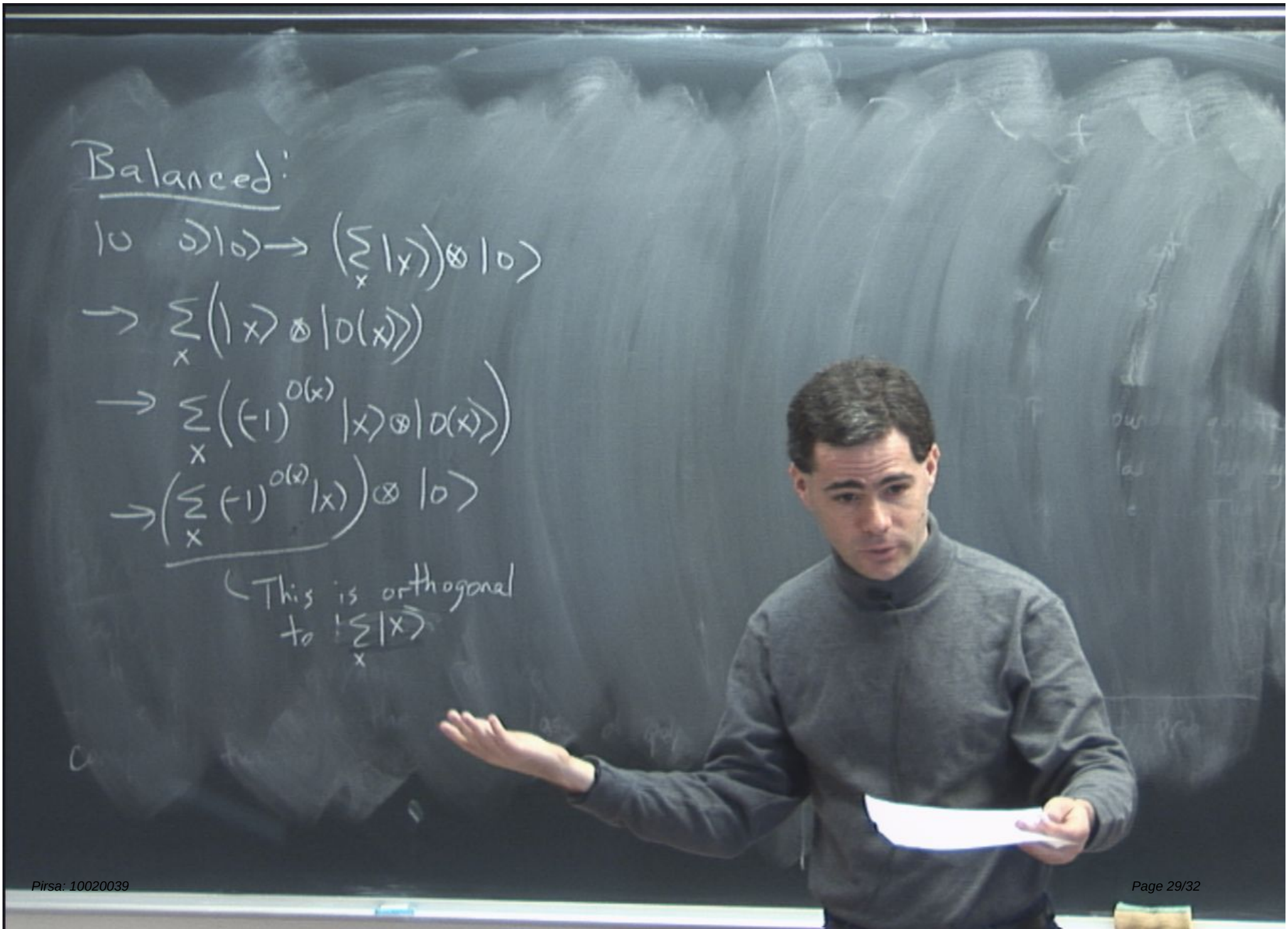
$$\to \left(\sum_x |x\rangle \otimes |O(x)\rangle\right) = \left(\sum_x |x\rangle\right)\otimes|O(0)\rangle$$

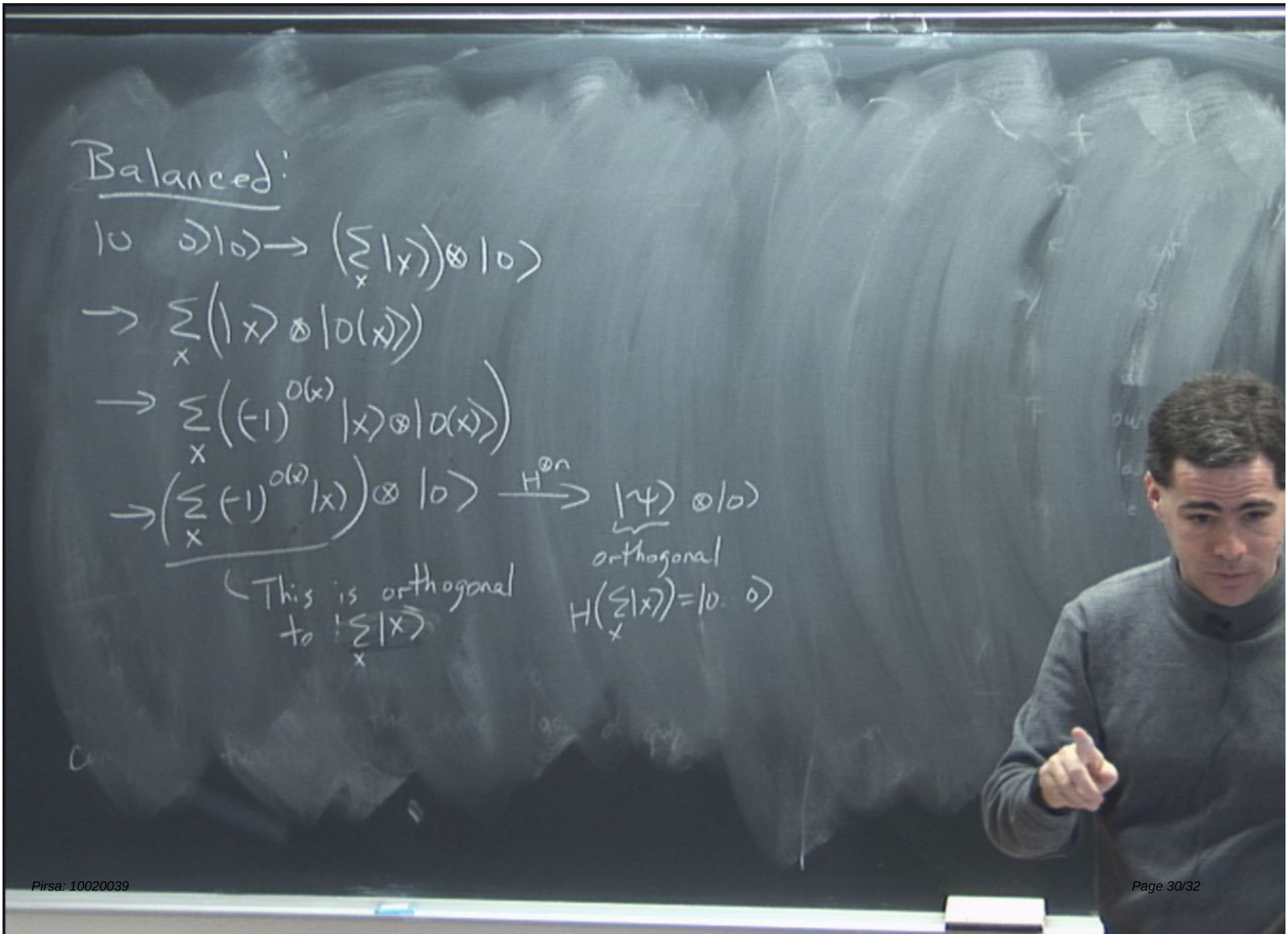
$$\to (-1)^{O(0)}\left(\sum_x |x\rangle\right)\otimes|O(0)\rangle$$

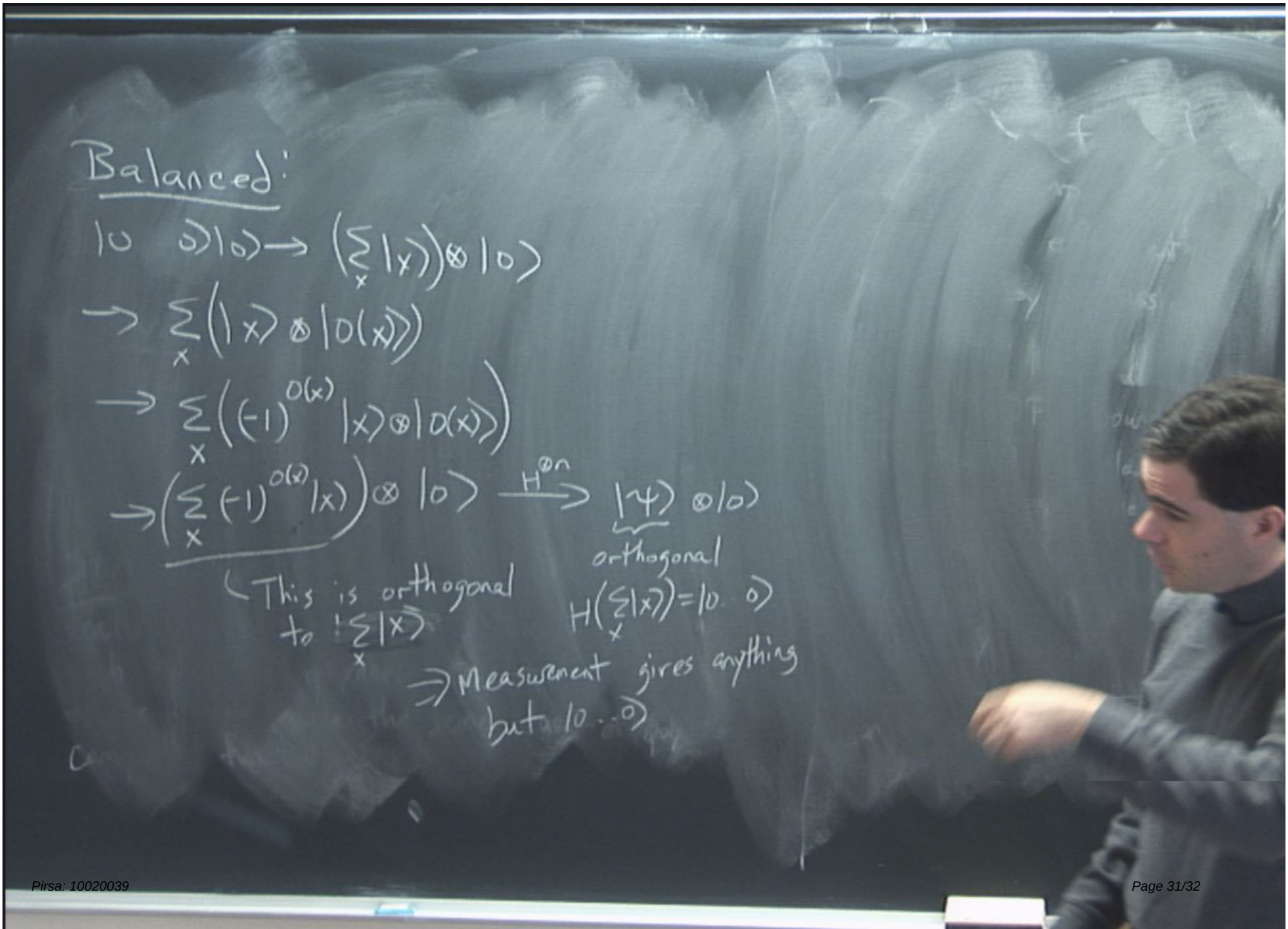
$$\to (-1)^{O(0)}\left(\sum_x |x\rangle\right)\otimes|0\rangle$$

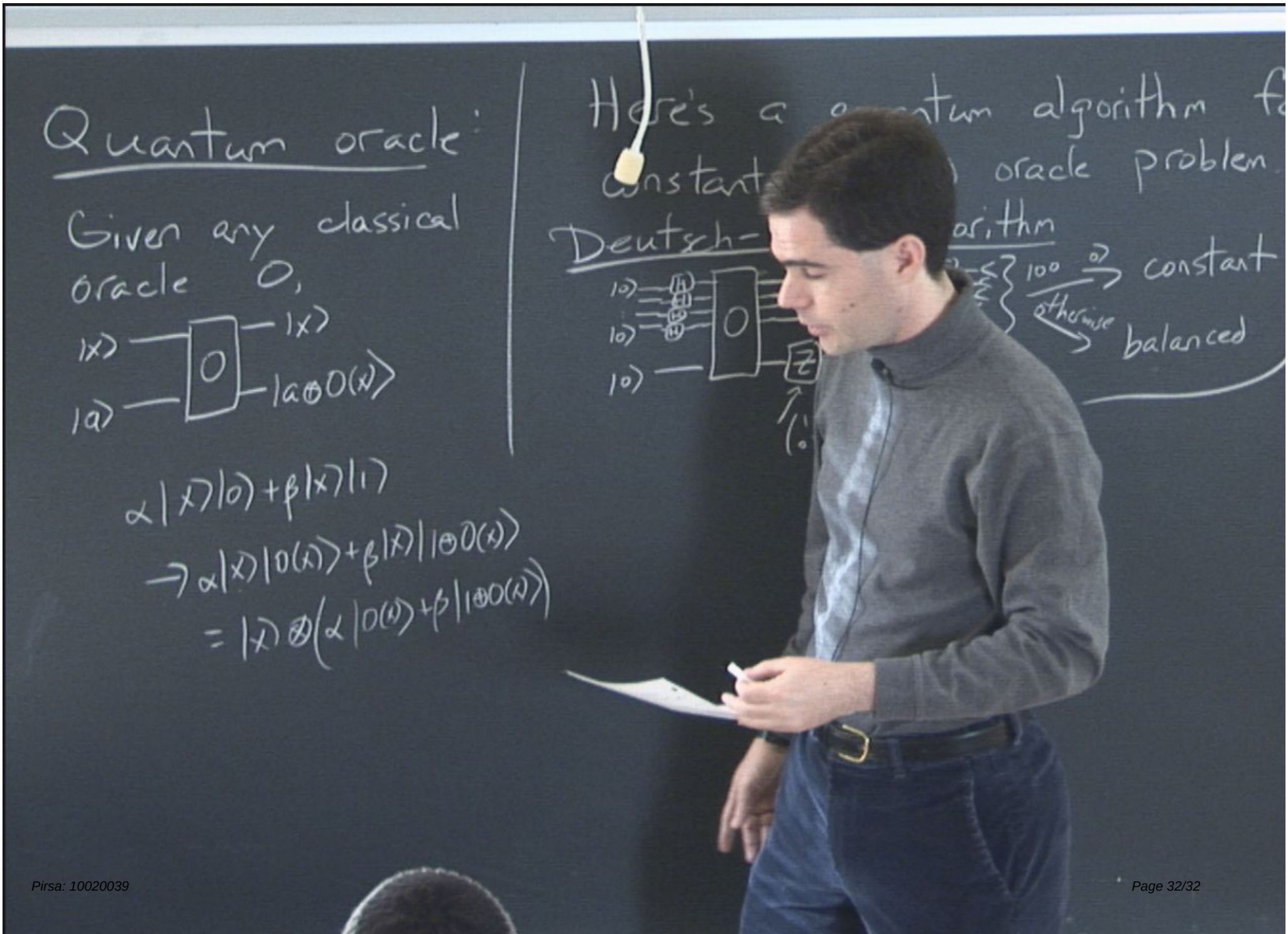
$$\to (-1)^{O(0)}|0\ldots0\rangle\otimes|0\rangle$$


Balanced:
$|0\ \ 0\rangle|0\rangle \rightarrow \left(\sum_x |x\rangle\right) \otimes |0\rangle$
$\rightarrow \sum_x \left(|x\rangle \otimes |0(x)\rangle\right)$
$\rightarrow \sum_x \left((-1)^{0(x)} |x\rangle \otimes |0(x)\rangle\right)$
$\rightarrow$


Balanced:
$|0\ \ 0\rangle|0\rangle \to \left(\sum_x |x\rangle\right) \otimes |0\rangle$
$\to \sum_x \left(|x\rangle \otimes |0(x)\rangle\right)$
$\to \sum_x \left((-1)^{0(x)} |x\rangle \otimes |0(x)\rangle\right)$
$\to \left(\sum_x (-1)^{0(x)} |x\rangle\right) \otimes |0\rangle$
This is orthogonal to $\sum_x |x\rangle$


Balanced:
$|0\ldots0\rangle|0\rangle \to \left(\sum_x |x\rangle\right)\otimes|0\rangle$
$\to \sum_x \left(|x\rangle \otimes |O(x)\rangle\right)$
$\to \sum_x \left((-1)^{O(x)}|x\rangle \otimes |O(x)\rangle\right)$
$\to \left(\sum_x (-1)^{O(x)}|x\rangle\right)\otimes|0\rangle \xrightarrow{H^{\otimes n}} |\psi\rangle \otimes |0\rangle$
orthogonal
This is orthogonal to $\sum_x |x\rangle$
$H\left(\sum_x |x\rangle\right) = |0\ldots0\rangle$


Balanced:
$|0\ \ldots\ 0\rangle|0\rangle \to \left(\sum_x |x\rangle\right) \otimes |0\rangle$
$\to \sum_x \left(|x\rangle \otimes |0(x)\rangle\right)$
$\to \sum_x \left((-1)^{0(x)} |x\rangle \otimes |0(x)\rangle\right)$
$\to \left(\sum_x (-1)^{0(x)} |x\rangle\right) \otimes |0\rangle \xrightarrow{H^{\otimes n}} |\psi\rangle \otimes |0\rangle$
orthogonal
This is orthogonal to $\sum_x |x\rangle$
$H\left(\sum_x |x\rangle\right) = |0\ \ldots\ 0\rangle$
$\Rightarrow$ Measurement gives anything but $|0\ldots 0\rangle$


Quantum oracle:
Given any classical oracle O,
|x⟩
|a⟩
O
|x⟩
|a⊕O(x)⟩
α|x⟩|0⟩+β|x⟩|1⟩
→ α|x⟩|O(x)⟩+β|x⟩|1⊕O(x)⟩
= |x⟩⊗(α|O(x)⟩+β|1⊕O(x)⟩)
Here's a quantum algorithm
constant
oracle problem
Deutsch-
orithm
|0⟩
|0⟩
|0⟩
O
Z
100
constant
otherwise
balanced