

Title: 13/14 PSI - Quantum Information Review - Lecture 5

Date: Feb 21, 2014 03:30 PM

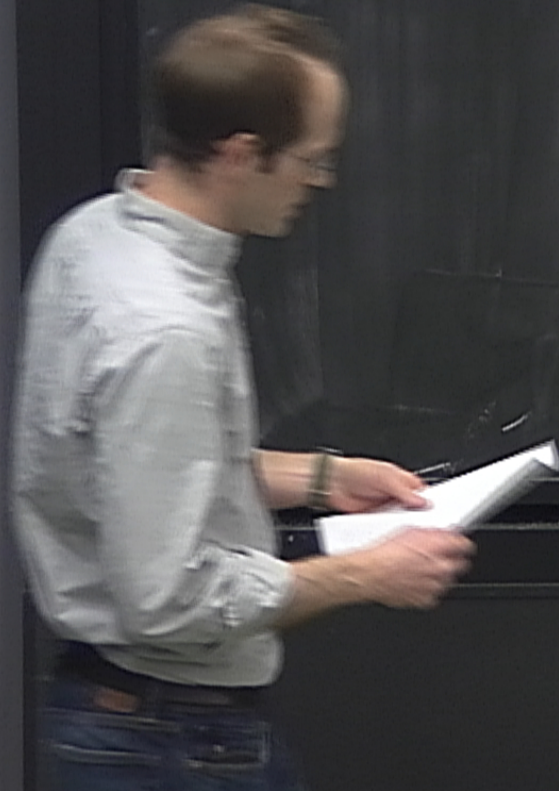
URL: <http://pirsa.org/14020153>

Abstract:



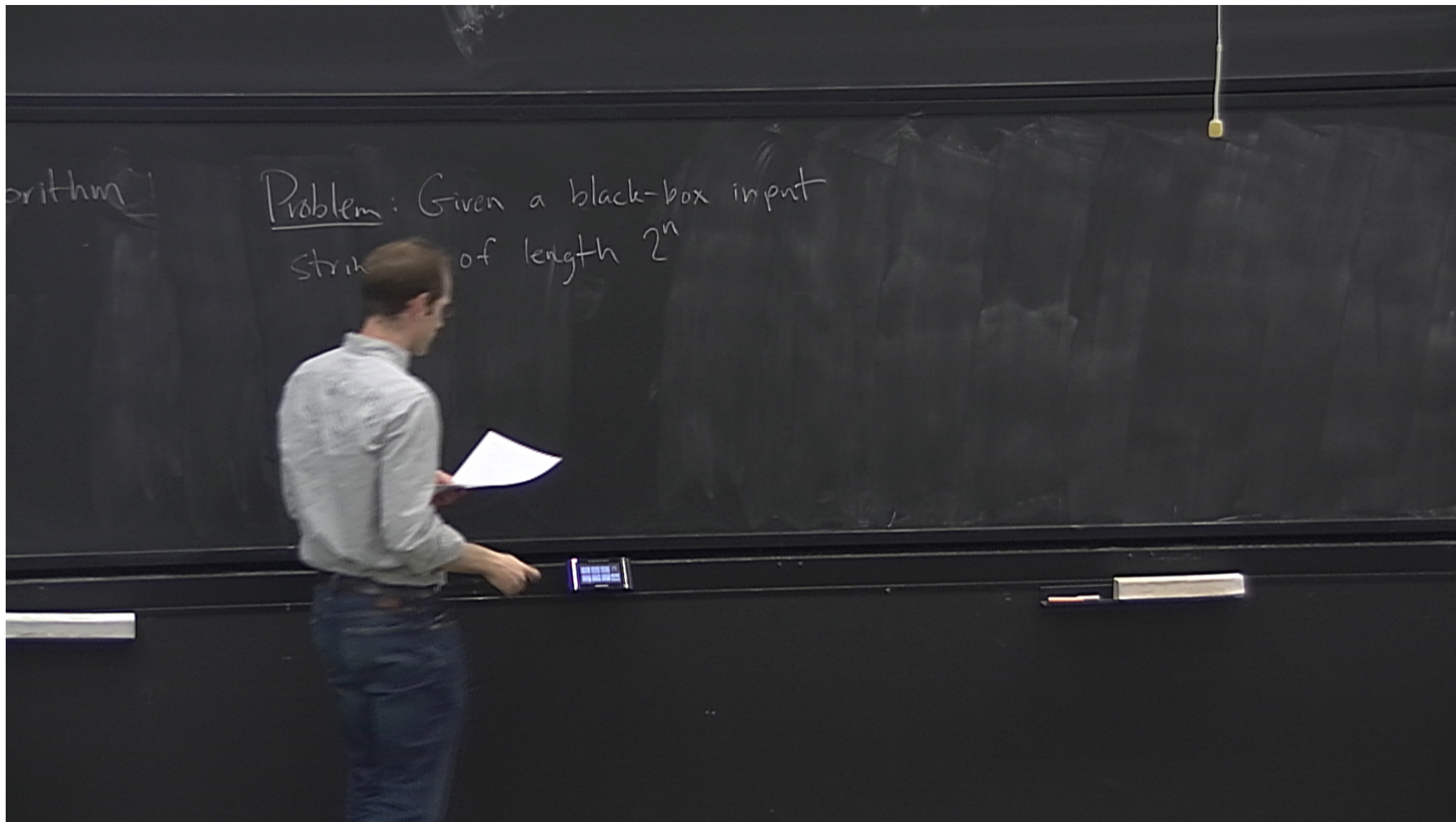


- Deutsch-Jozsa algorithm
- Simon's problem



- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given



- Deutsch-Jozsa algorithm
- Simon's problem

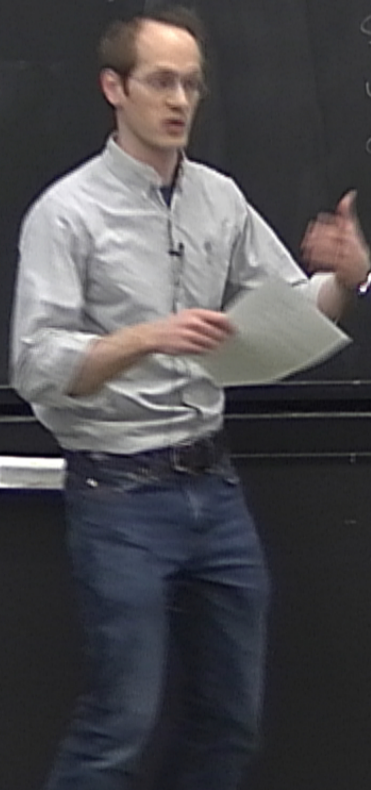
Problem: Given a black-box input string x length 2^n , decide whether x is constant or not.

- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).

- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).

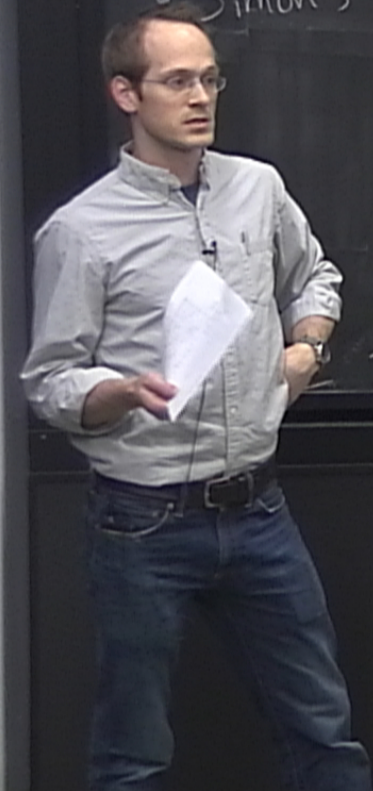


- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).

- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).



- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).

- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).

- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (0...0 or 1...1) or balanced (equal # of 0s and 1s).

- Deutsch-Jozsa algorithm
- Simon's problem

Problem: Given a black-box input string x of length 2^n , decide whether x is constant ($0 \dots 0$ or $1 \dots 1$) or balanced (equal # of 0s and 1s).

Deterministic classical query complexity is $\frac{2^n}{2} + 1$.

Problem: Given a black-box input string x of length 2^n , decide whether x is constant (all 0s or all 1s) or balanced (equal number of 0s and 1s).

Deterministic classical complexity is $\frac{2^n}{2} + 1$.

Quantum black box:

$|i, z\rangle$

thm
Problem: Given a black-box input string x of length 2^n , decide whether x is constant ($0 \dots 0$ or $1 \dots 1$) or balanced (equal # of 0s and 1s).

Deterministic classical query complexity is $\frac{2^n}{2} + 1$.

Quantum black box:

$$|i, z\rangle \mapsto |i, z \oplus x_i\rangle$$

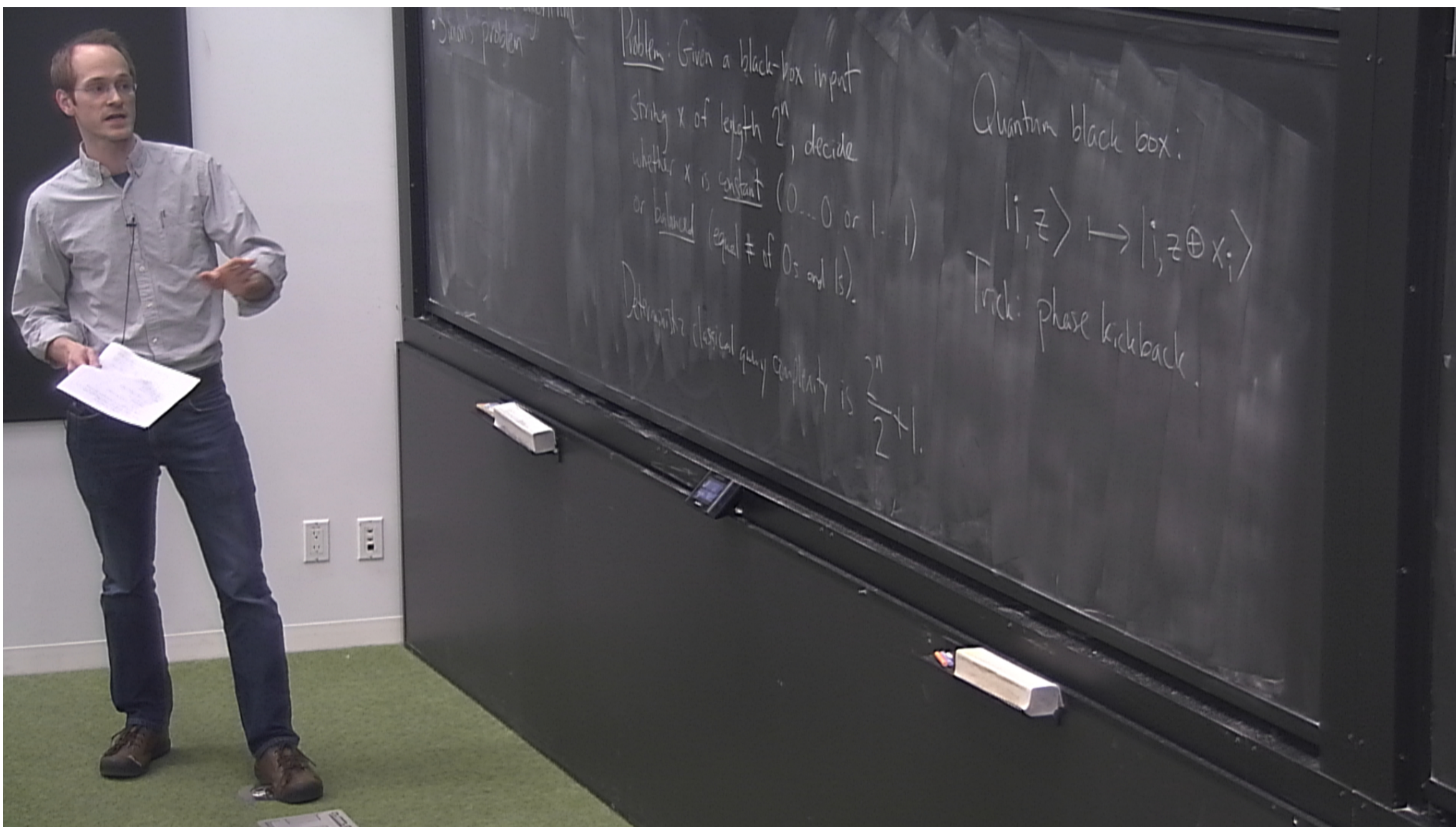
thm
Problem: Given a black-box input string x of length 2^n , decide whether x is constant ($0 \dots 0$) or balanced (equal # of 0s and 1s).

Deterministic classical query complexity $\Omega(2^n)$.

Quantum black box:

$$|i, z\rangle \mapsto |i, z \oplus x_i\rangle$$

Trick: phase kickback.



input

decide

- 0 or 1... 1)

and 1s).

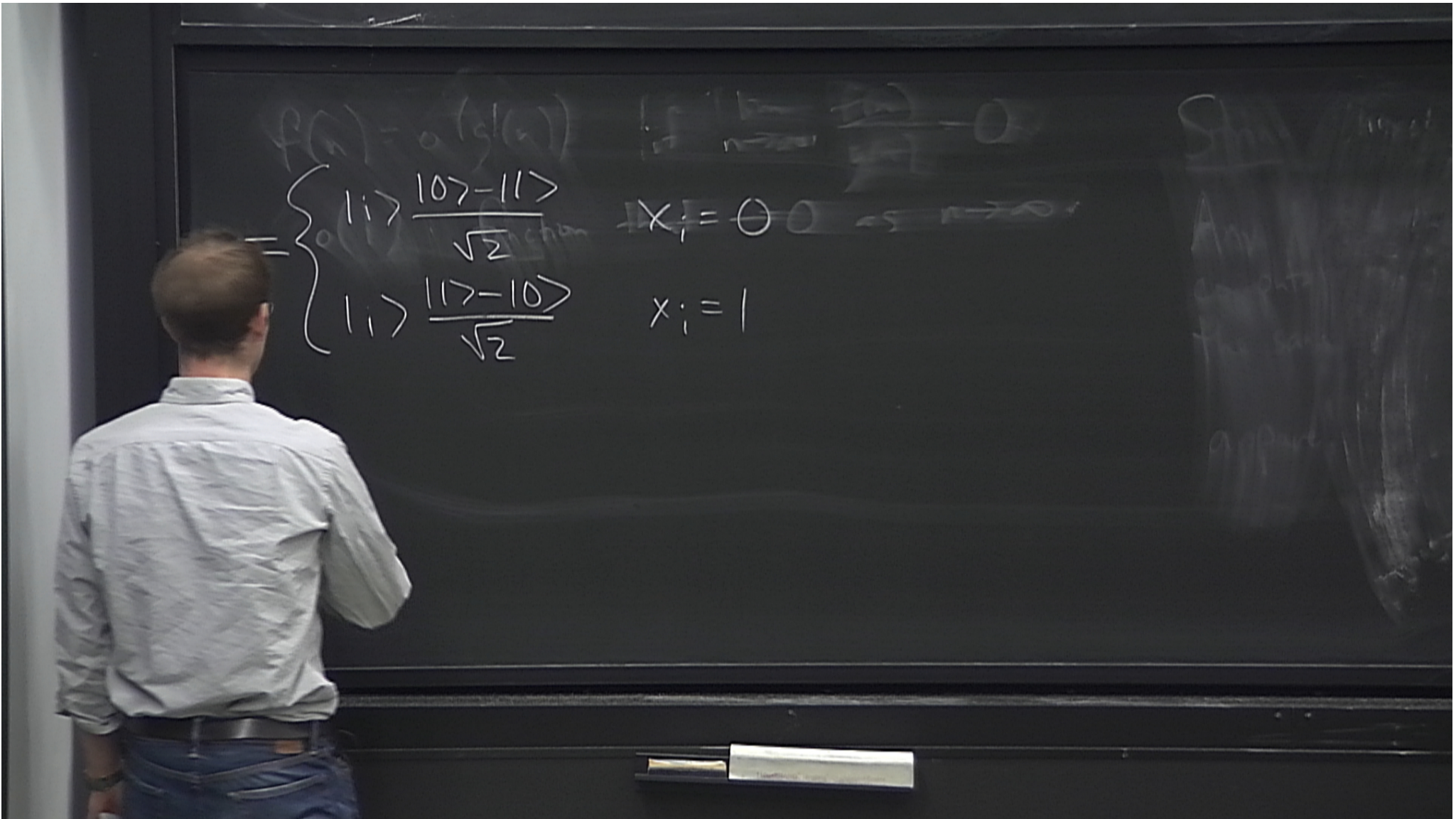
ity is $\frac{2^n}{2} + 1$.

Quantum black box:

$$|i, z\rangle \mapsto |i, z \oplus x_i\rangle$$

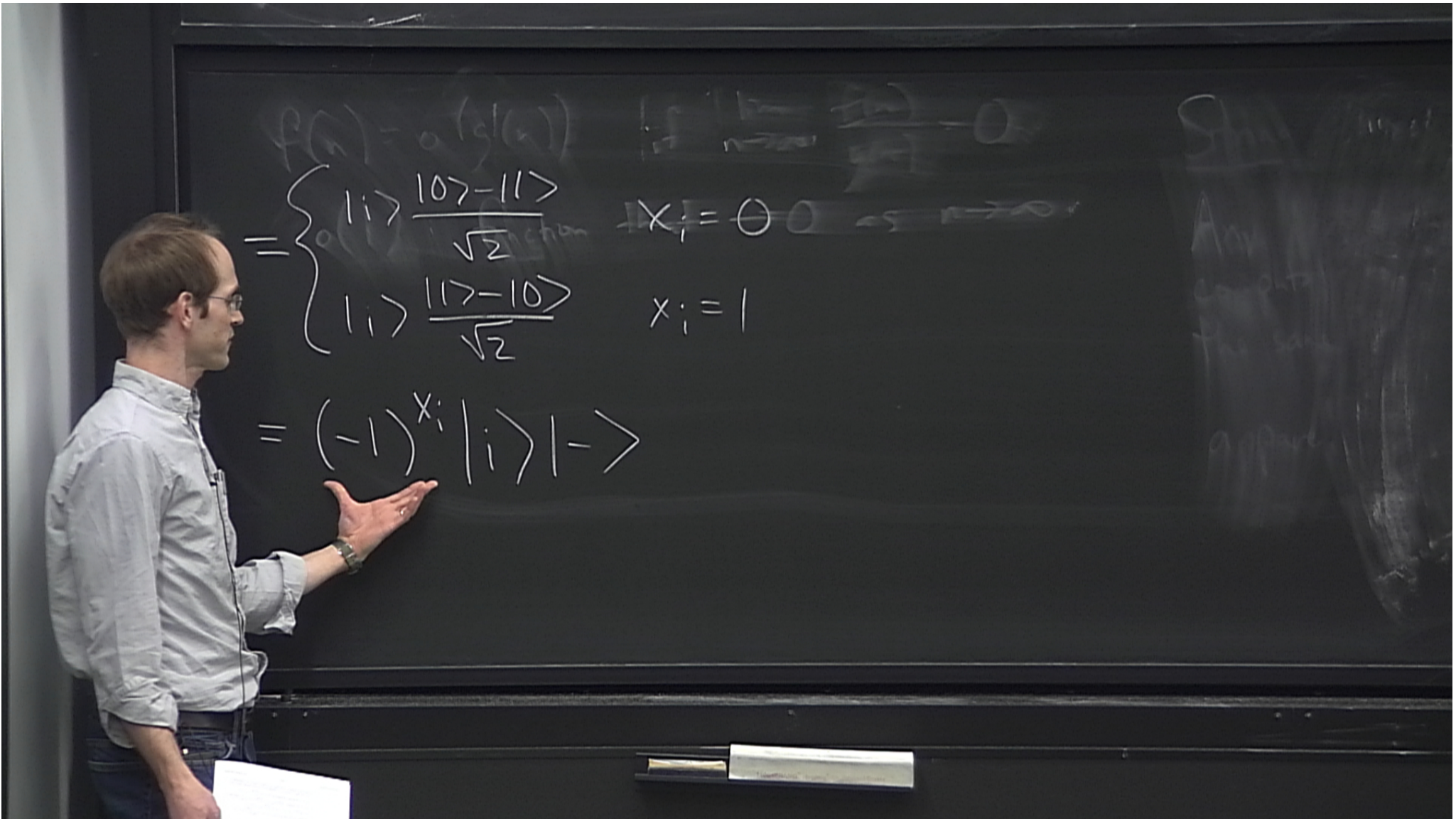
Trick: phase kickback.

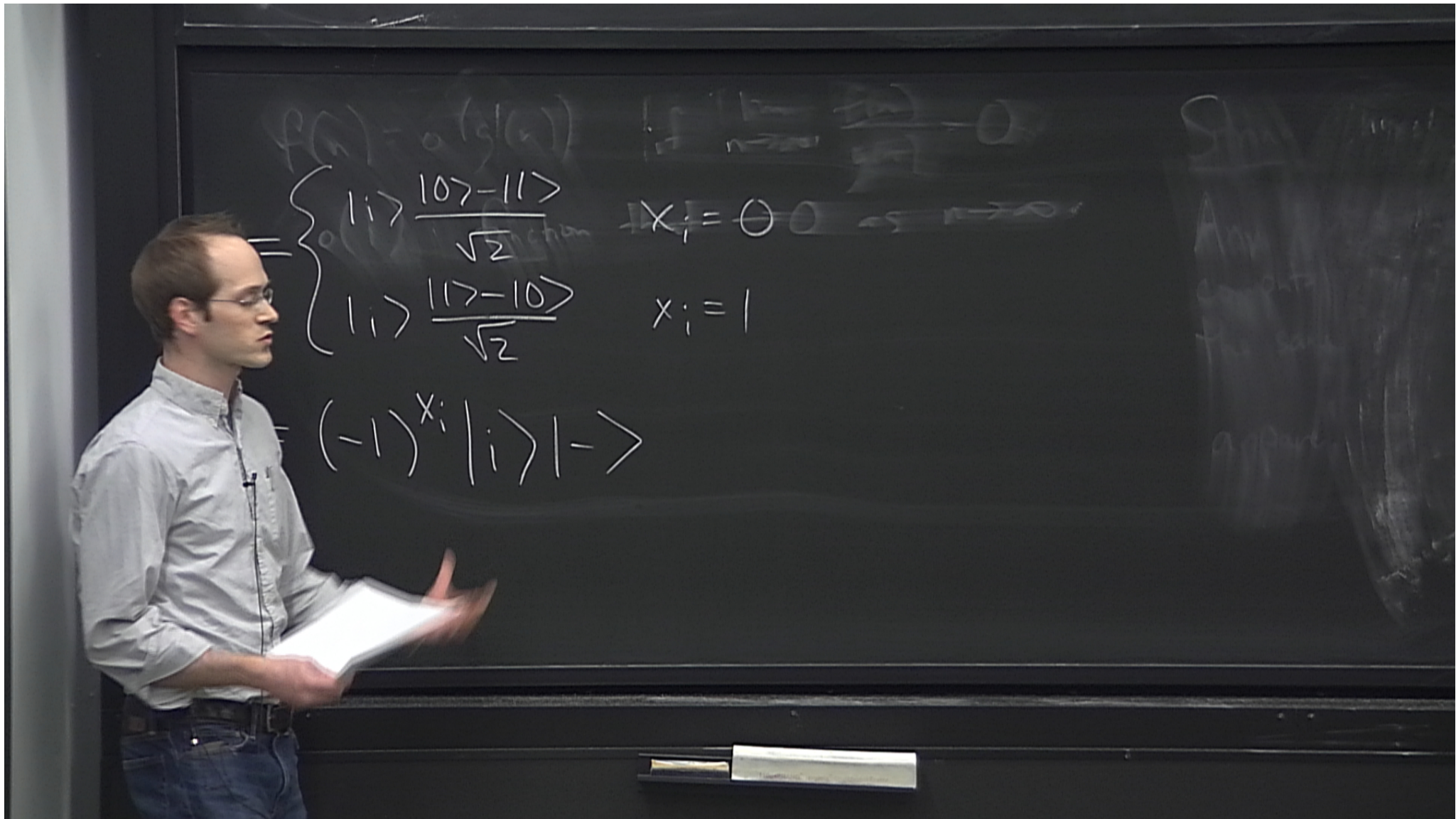
$$|i\rangle|-\rangle = |i\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$



$$= \begin{cases} |i\rangle \frac{107-117}{\sqrt{2}} & x_i = 0 \\ |i\rangle \frac{117-107}{\sqrt{2}} & x_i = 1 \end{cases}$$

$$= (-1)^{x_i} |i\rangle |-\rangle$$





$$f(n) = o(g(n)) \quad \text{Algorithm:}$$

$$= \begin{cases} |i\rangle \frac{107-11}{\sqrt{2}} & x_i = 0 \\ |i\rangle \frac{117-10}{\sqrt{2}} & x_i = 1 \end{cases}$$

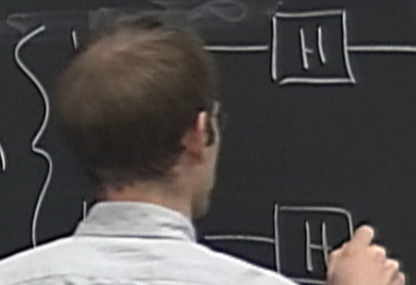
$$= (-1)^{x_i} |i\rangle |-\rangle$$

$$f(n) = O(g(n)) \quad \text{if } n \geq n_0$$

$$= \begin{cases} |i\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} & x_i = 0 \\ |i\rangle \frac{|1\rangle - |0\rangle}{\sqrt{2}} & x_i = 1 \end{cases}$$

$$= (-1)^{x_i} |i\rangle |-\rangle$$

Algorithm:



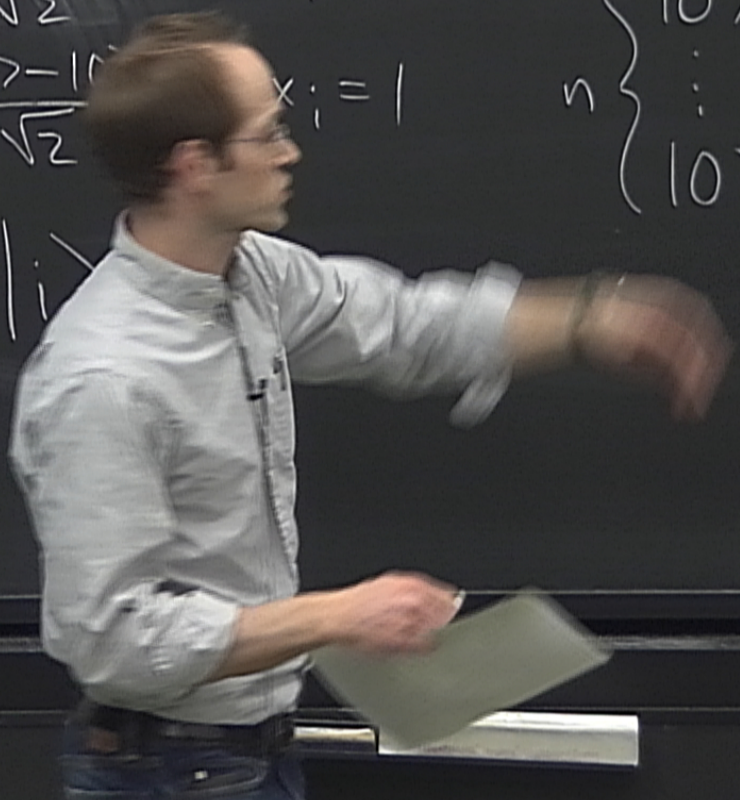
$$f(n) = o(g(n)) \quad \text{if } n \rightarrow \infty$$

$$= \begin{cases} |i\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} & x_i = 0 \\ |i\rangle \frac{|1\rangle - |0\rangle}{\sqrt{2}} & x_i = 1 \end{cases}$$

$$= (-1)^{x_i} |i\rangle$$

Algorithm:

$$\begin{cases} |0\rangle - \boxed{H} - \\ \vdots \\ |0\rangle - \boxed{H} - \end{cases}$$



$$f(n) = O(g(n)) \quad \text{if } n \geq n_0$$

Algorithm:

$$= \begin{cases} |i\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} & x_i = 0 \\ |i\rangle \frac{|1\rangle - |0\rangle}{\sqrt{2}} & x_i = 1 \end{cases}$$

$$= (-1)^{x_i} |i\rangle |-\rangle$$

$n \begin{cases} |0\rangle \rightarrow \text{Ans} \\ \vdots \\ |0\rangle \rightarrow \text{query} \end{cases}$

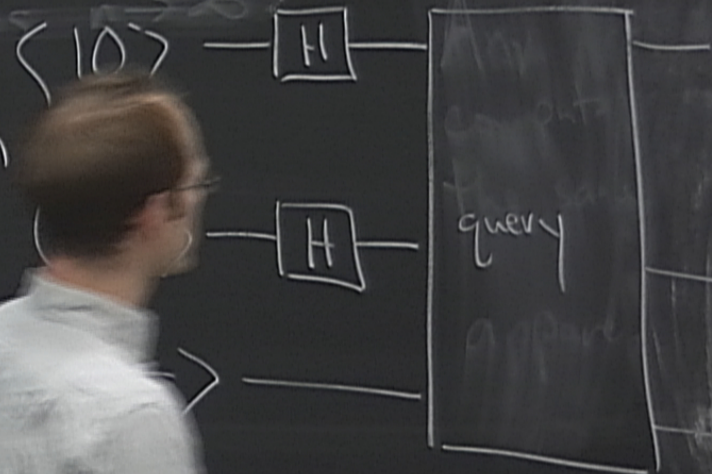
$$= \begin{cases} |i\rangle \frac{107-11}{\sqrt{2}} \\ |i\rangle \frac{117-10}{\sqrt{2}} \end{cases}$$

$$= (-1)^{x_i} |i\rangle |-\rangle$$

$$x_i = 0$$

$$x_i = 1$$

Algorithm:



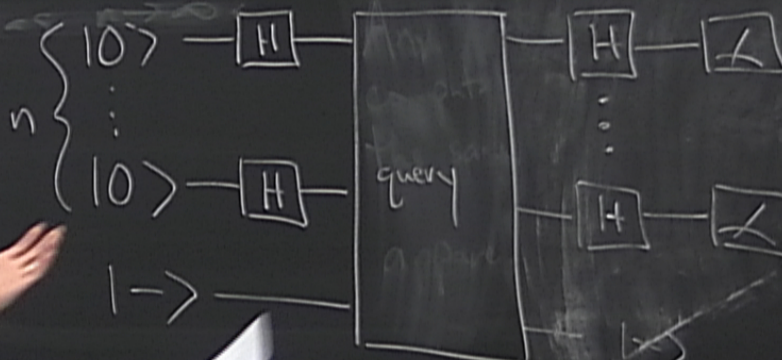
Deterministic classical query complexity is 2^n

Algorithm:

$$= \begin{cases} |i\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} \\ |1\rangle \frac{|1\rangle - |0\rangle}{\sqrt{2}} \end{cases}$$

$$x_i = 0$$

x



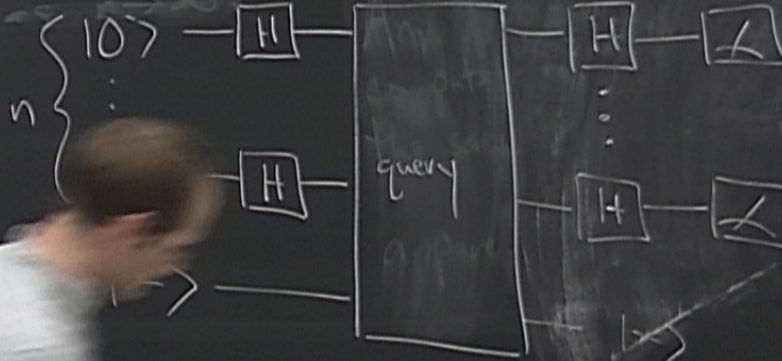
$$= (-1)^{x_i} |i\rangle |-\rangle$$

Deterministic classical query complexity is $\frac{n+1}{2}$

Algorithm:

$$= \begin{cases} |i\rangle \frac{|0\rangle - |1\rangle}{\sqrt{2}} & x_i = 0 \\ |i\rangle \frac{|1\rangle - |0\rangle}{\sqrt{2}} & x_i = 1 \end{cases}$$

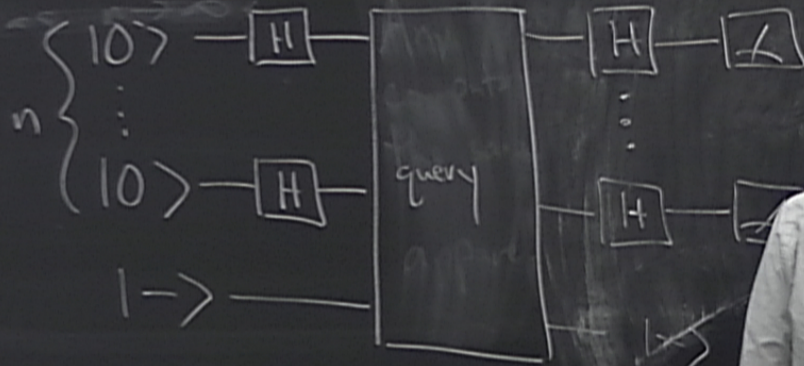
$$= (-1)^{x_i} |i\rangle |-\rangle$$



Deterministic classical query complexity is $\frac{n}{2} + 1$.

Algorithm:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$



$$|+\rangle \xrightarrow{H} |++\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$\otimes^n |0\rangle$$

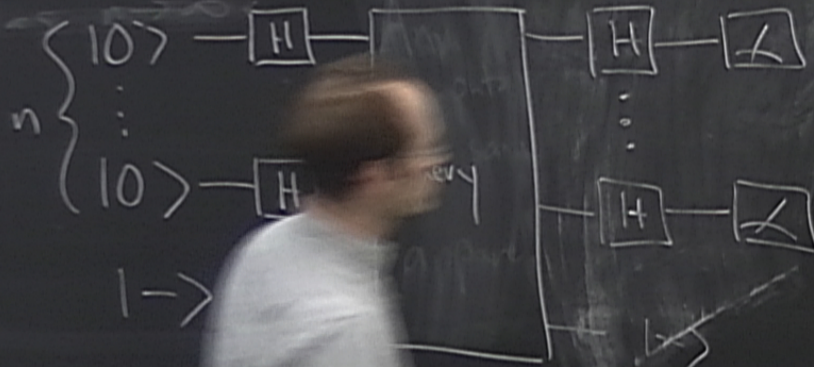
Deterministic classical query complexity is $\frac{n}{2} + 1$.

Algorithm:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$|0\rangle \xrightarrow{H} |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

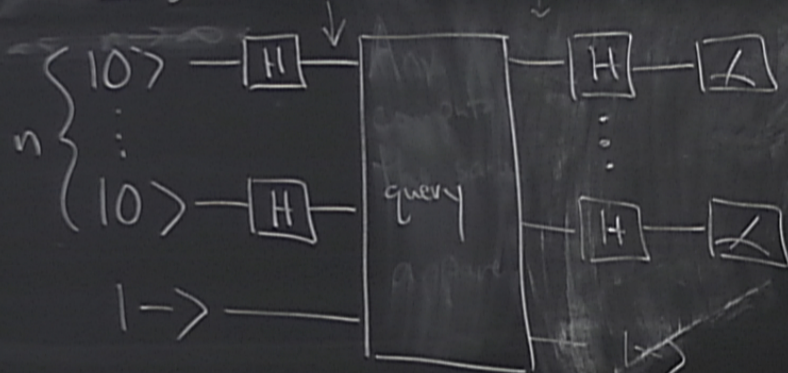
$$H^{\otimes n} |0\rangle^{\otimes n}$$



Deterministic classical query complexity is $\frac{n}{2} + 1$.

$$H \rightarrow \frac{|1\rangle|x_1\rangle - |1\rangle|x_2\rangle}{\sqrt{2}}$$

Algorithm:



$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$|0\rangle \xrightarrow{H} |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

$$H^{\otimes n} |0\rangle^{\otimes n} = |+\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$$

$$H|a\rangle = \frac{1}{\sqrt{2}} \sum_{b=0}^1 (-1)^{ab} |b\rangle$$

$$H^{\otimes n} |i\rangle = \frac{1}{\sqrt{2^n}} \sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle$$

So the final state before the measurement is

$$\frac{1}{\sqrt{2^n}} \sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle$$

$$H|a\rangle = \frac{1}{\sqrt{2}} \sum_{b=0}^1 (-1)^{a \cdot b} |b\rangle$$

$$|i\rangle = \frac{1}{\sqrt{2^n}} \sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle$$

So the final state before the measurement is

$$\frac{1}{2^n} \sum_{i,j \in \{0,1\}^n} (-1)^{x_i + i \cdot j} |j\rangle$$

$$\text{Pr}(j) = \frac{1}{2^{2n}} \left| \sum_{i \in \{0,1\}^n} (-1)^{x_i + i \cdot j} \right|^2$$

$$H|a\rangle = \frac{1}{\sqrt{2}} \sum_{b=0}^1 (-1)^{ab} |b\rangle$$

$$H^{\otimes n} |a\rangle = \frac{1}{\sqrt{2^n}} \sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle$$

So the final state before the measurement is

$$\frac{1}{2^n} \sum_{i,j \in \{0,1\}^n} (-1)^{x_i + i \cdot j} |j\rangle$$

$$\Pr(j) = \frac{1}{2^{2n}} \left| \sum_{i \in \{0,1\}^n} (-1)^{x_i + i \cdot j} \right|^2$$

$$\mapsto \frac{1}{\sqrt{2^n}} \sum_{i \in \{0,1\}^n} (-1)^{x_i} |i\rangle$$

So the final state before the measurement is

$$\frac{1}{2^n} \sum_{i,j \in \{0,1\}^n} (-1)^{x_i + i \cdot j} |j\rangle$$

$$Pr(j) = \frac{1}{2^{2n}} \left| \sum_{i \in \{0,1\}^n} (-1)^{x_i + i \cdot j} \right|^2$$

in particular,

$$Pr(0^n) = \frac{1}{2^{2n}} \left| \sum_{i \in \{0,1\}^n} (-1)^{x_i} \right|^2$$

but — the randomized query complexity is $\Theta(1)$



but — the randomized query complexity is $\Theta(1)$
Better separations are possible

Simon's problem

Problem: given black-box function $f: \{0,1\}^n$



but — the randomized query complexity is $\Theta(1)$

Better separations are possible

Simon's problem

Problem: given black-box function $f: \{0,1\}^n \rightarrow S$

Promise: $f(x)$

same
input
get