Introduction to Quantum Information Processing
QIC 710 / CS 768 / PH 767 / CO 681 / AM 871

Lecture 7 (2017)

Jon Yard
QNC 3126
jyard@uwaterloo.ca
http://math.uwaterloo.ca/~jyard/qic710

Continuing with the QFT for $m = 2^n$

Quantum Fourier transform

$$F_m = \frac{1}{\sqrt{m}} \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \cdots & \omega^{m-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \cdots & \omega^{2(m-1)} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \cdots & \omega^{3(m-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{m-1} & \omega^{2(m-1)} & \omega^{3(m-1)} & \cdots & \omega^{(m-1)^2} \end{pmatrix}$$

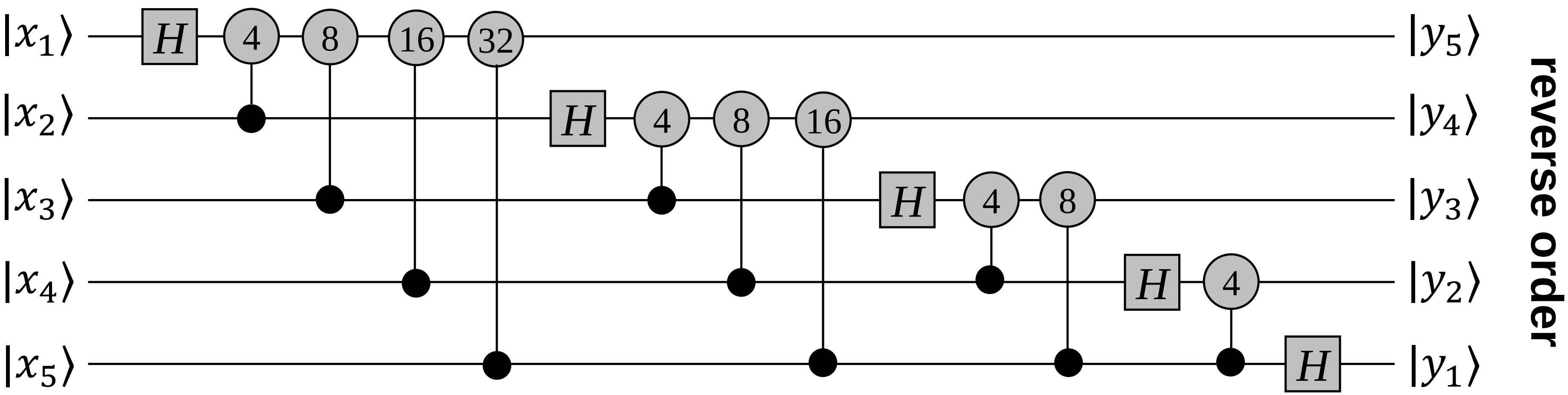
where $\omega = e^{2\pi i/m}$ (for n qubits, $m = 2^n$).

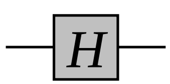
This is unitary and generalizes the Hadmard transform $F_2 = H$.

The quantum Fourier transform is an important component of several interesting quantum algorithms.

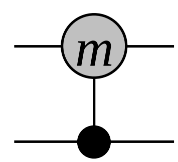
Computing the QFT for $m = 2^n$ (1)

Quantum circuit for F_{32}



Gates: 

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



$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{2\pi i/m} \end{pmatrix}$

Controlled phase gate
(who controls who?)

F_{2^n} costs $O(n^2)$ gates.

Computing the QFT for $m = 2^n$ (2)

One way on seeing why this circuit works is to show:

1. The output of the circuit (before reversing the qubits) is

$$(|0\rangle + e^{2\pi i(0.x_1 x_2 \cdots x_n)}|1\rangle)(|0\rangle + e^{2\pi i(0.x_2 \cdots x_n)}|1\rangle) \cdots (|0\rangle + e^{2\pi i(0.x_n)}|1\rangle)$$

2. After reversing the qubits,

$$F_{2^n}|x_1 x_2 \cdots x_n\rangle =$$

$$(|0\rangle + e^{2\pi i(0.x_n)}|1\rangle) \cdots (|0\rangle + e^{2\pi i(0.x_2 \cdots x_n)}|1\rangle)(|0\rangle + e^{2\pi i(0.x_1 x_2 \cdots x_n)}|1\rangle)$$

$$= \sum_{y=0}^{2^n - 1} \omega^{xy}|y\rangle$$

Hidden Subgroup Problem framework

Hidden subgroup problem (commutative version)

Let G be a known group and $H \subset G$ be an unknown subgroup

Let $f: G \to T$ have the property $f(x) = f(y)$ iff $x - y \in H$
(i.e. x and y are in the same ***coset*** of H)

Problem: given a black-box for computing f, determine H

Example 1: $G = \mathbb{F}_2^n$ (the additive group) and $H = \{0, r\}$

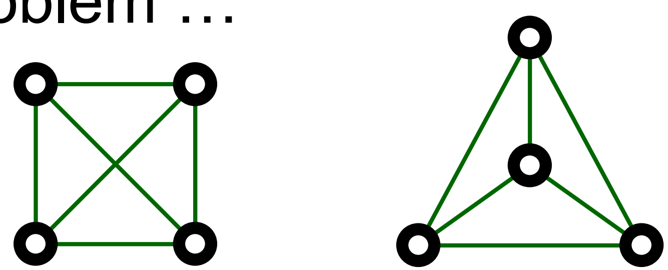
Example 2: $G = \left(\mathbb{Z}_{p-1}\right)^2$ and
$H = \mathbb{Z}_{p-1}(r, 1) = \{(0,0), (r, 1), (2r, 2), \dots, ((p - 2)r, p - 2)\}$

Example 3: $G = \mathbb{Z}$ and $H = r\mathbb{Z}$ (Shor's factoring algorithm was originally approached this way. A complication that arises is that $\mathbb{Z}$ is infinite. We'll use a different approach)

Hidden subgroup problem (noncommutative version)

Example 4: $G = S_n$ (the symmetric group, consisting of all permutations on n objects—which is not commutative) and H is any subgroup of G (and we use ***left*** cosets throughout)

A quantum algorithm for this instance of HSP ***would*** lead to an efficient quantum algorithm for the graph isomorphism problem …



…***but still,*** no polynomial-time quantum has been found for this instance of HSP, despite significant effort by many people. **However**, Babai recently claimed (then retracted, then unretracted) a quasi-polynomial-time ($\exp(O(\text{polylog}(n)))$) *classical* algorithm. Still not peer-reviewed…

Eigenvalue estimation problem (a.k.a. phase estimation)

Note: this will lead to a factoring algorithm similar to Shor's

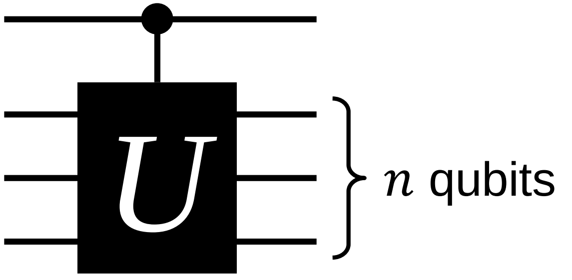
A simplified example

U is an unknown unitary operation on n qubits

$|\psi\rangle$ is an eigenvector of U, with eigenvalue $\lambda = +1$ or -1

Input: a black-box for a controlled-U

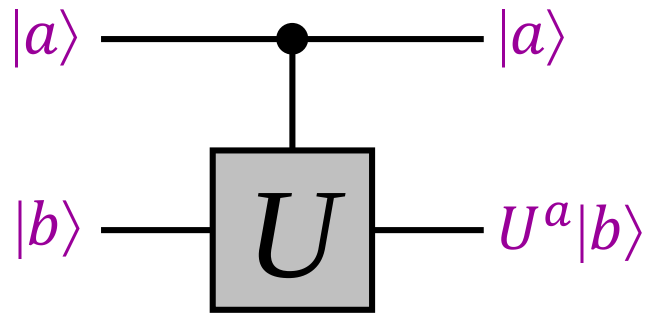
and a copy of the state $|\psi\rangle$



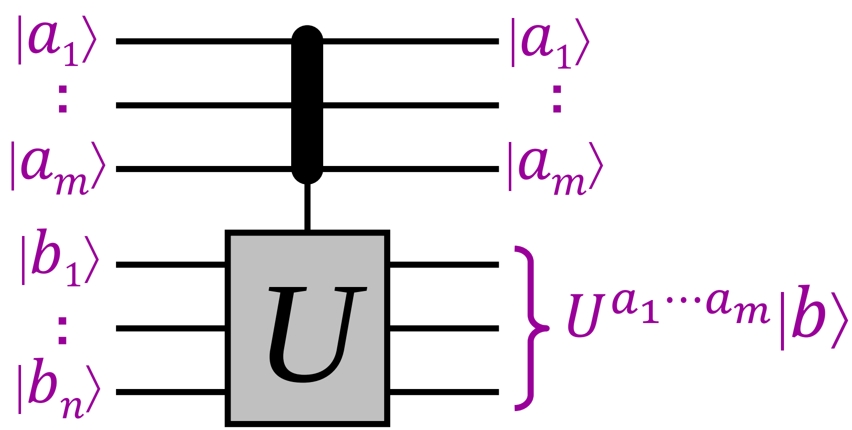
Output: the eigenvalue λ

Exercise: solve this making a single query to the controlled-U

Generalized controlled-U gates



$$\begin{bmatrix} I & 0 \\ 0 & U \end{bmatrix}$$



$$\begin{bmatrix} I & 0 & 0 & \cdots & 0 \\ 0 & U & 0 & \cdots & 0 \\ 0 & 0 & U^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & U^{2^m-1} \end{bmatrix}$$

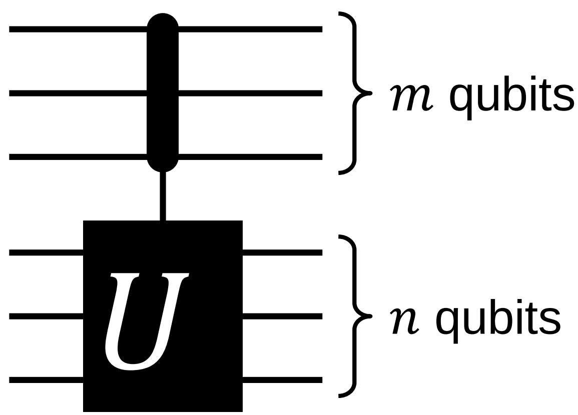
Example: $|1101\rangle|0101\rangle \mapsto |1101\rangle U^{13}|0101\rangle$

Eigenvalue estimation problem

U is a unitary operation on n qubits

$|\psi\rangle$ is an eigenvector of U, with eigenvalue $e^{2\pi i\phi}$ $(0 \leq \phi < 1)$

Input: black-box for

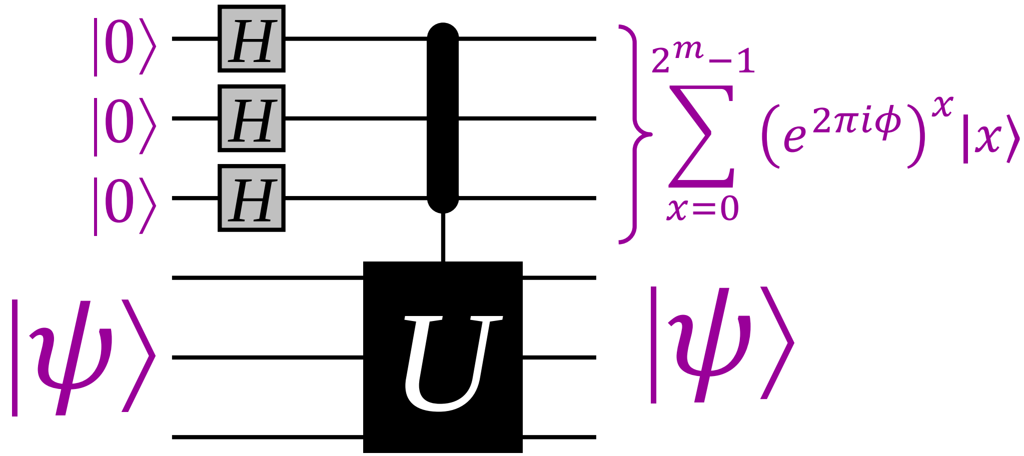


and a copy of $|\psi\rangle$

Output: ϕ (m-bit approximation)

Algorithm for eigenvalue estimation (1)

Starts off as:



$|00\cdots 0\rangle|\psi\rangle$

$|a\rangle|b\rangle \to |a\rangle U^a|b\rangle$

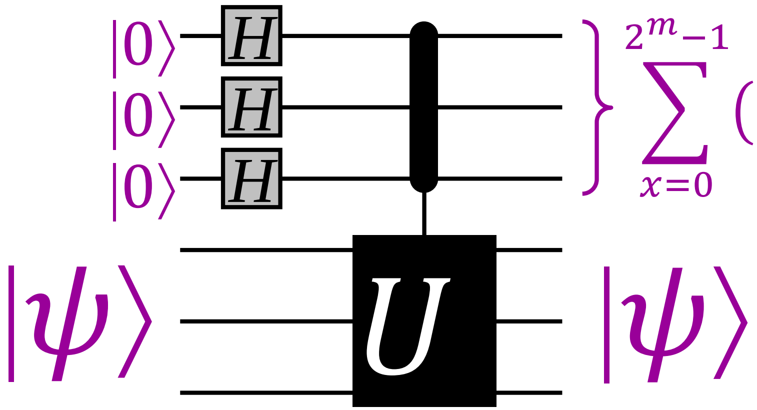
$\mapsto (|0\rangle + |1\rangle)(|0\rangle + |1\rangle)\cdots(|0\rangle + |1\rangle)|\psi\rangle$

$= (|000\rangle + |001\rangle + |010\rangle + |011\rangle + \cdots + |111\rangle)|\psi\rangle$

$= (|0\rangle + |1\rangle + |2\rangle + |3\rangle + \cdots + |2^{m-1}\rangle)|\psi\rangle$

$\mapsto (|0\rangle + e^{2\pi i\phi}|1\rangle + (e^{2\pi i\phi})^2|2\rangle + (e^{2\pi i\phi})^3|3\rangle + \cdots + (e^{2\pi i\phi})^{2^m-1}|2^m-1\rangle)\,|\psi\rangle$

Algorithm for eigenvalue estimation (2)



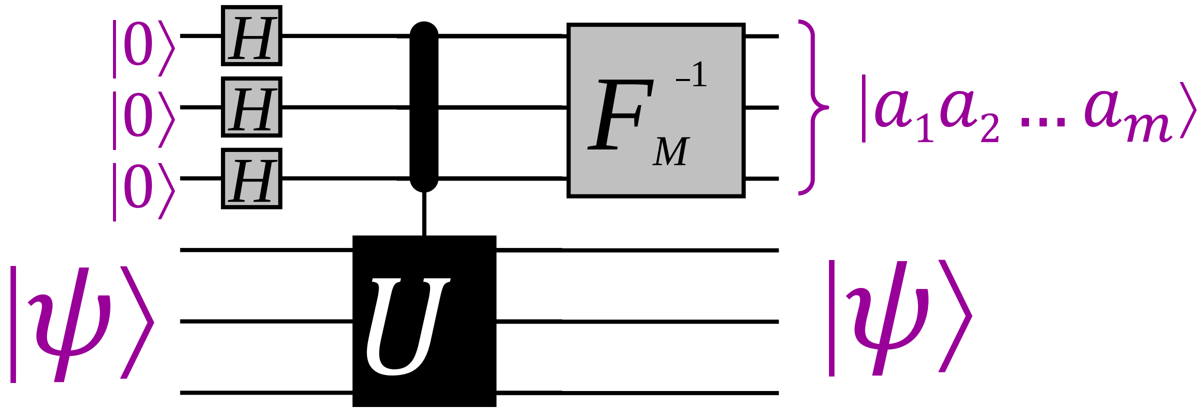
$$\sum_{x=0}^{2^m-1} \left(e^{2\pi i\phi}\right)^x |x\rangle$$

Recall that $F_m|a_1 a_2 \cdots a_m\rangle = \sum_{x=0}^{2^m-1} \left(e^{2\pi i(0.a_1 a_2 \cdots a_m)}\right)^x |x\rangle$

$$F_m^{-1} = \frac{1}{\sqrt{m}} \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^{-1} & \omega^{-2} & \omega^{-4} & \cdots & \omega^{-(m-1)} \\ 1 & \omega^{-2} & \omega^{-4} & \omega^{-6} & \cdots & \omega^{-2(m-1)} \\ 1 & \omega^{-3} & \omega^{-6} & \omega^{-9} & \cdots & \omega^{-3(m-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{-(m-1)} & \omega^{-2(m-1)} & \omega^{-3(m-1)} & \cdots & \omega^{-(m-1)^2} \end{pmatrix}$$

Therefore, when $\phi = 0.a_1 a_2 \cdots a_m$, applying the ***inverse*** of F_m yields ϕ (digits)

Algorithm for eigenvalue estimation (3)



If $\phi = 0.a_1 a_2 \cdots a_m$ then the above procedure yields $|a_1 a_2 \dots a_m\rangle$ (from which ϕ can be deduced exactly)

But what ϕ if is not of this nice form?

Example: $\phi = \frac{1}{3} = 0.01010101010101 \dots$

Algorithm for eigenvalue estimation (4)

What if ϕ is not of the nice form $\phi = 0.a_1 a_2 \cdots a_m$?

Example: $\phi = \frac{1}{3} = 0.010101010101010101\ldots$

Let's calculate what the previously-described procedure does:

Let $a/2^m = 0.a_1 a_2 \cdots a_m$ be an m-bit approximation of ϕ, in the sense that $\phi = a/2^m + \delta$, where $|\delta| \leq 1/2^{m+1}$

$$F_m^{-1} \sum_{x=0}^{2^m-1} \left(e^{2\pi i\phi}\right)^x |x\rangle = \frac{1}{2^m} \sum_{y=0}^{2^m-1} \sum_{x=0}^{2^m-1} e^{-2\pi ixy/2^m} e^{2\pi i\phi x} |y\rangle$$

$$= \frac{1}{2^m} \sum_{y=0}^{2^m-1} \sum_{x=0}^{2^m-1} e^{-2\pi ixy/2^m} e^{2\pi i\left(\frac{a}{2^m}+\delta\right)x} |y\rangle$$

$$= \frac{1}{2^m} \sum_{y=0}^{2^m-1} \sum_{x=0}^{2^m-1} e^{2\pi ix(a-y)/2^m} e^{2\pi i\delta x} |y\rangle$$

What is the amplitude of $|a_1 a_2 \ldots a_m\rangle$?

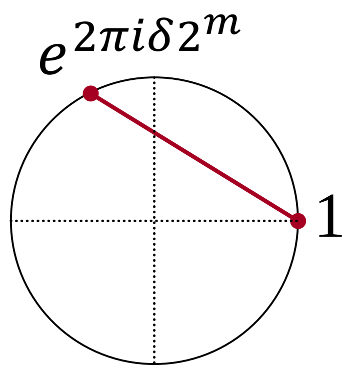
Algorithm for eigenvalue estimation (5)

State is: $\frac{1}{2^m}\sum_{y=0}^{2^m-1}\sum_{x=0}^{2^m-1} e^{2\pi i x(a-y)/2^m} e^{2\pi i \delta x}|y\rangle$

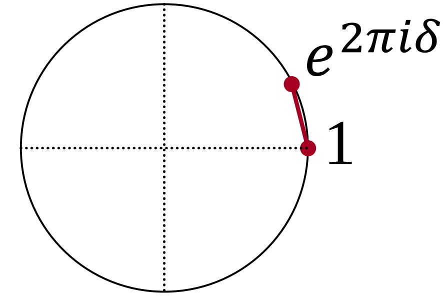
geometric series!

The amplitude of $|y\rangle$, for $y = a$ is $\frac{1}{2^m}\sum_{x=0}^{2^m-1} e^{2\pi i \delta x} = \frac{1}{2^m}\frac{1-\left(e^{2\pi i \delta}\right)^{2^m}}{1-e^{2\pi i \delta}}$

Numerator:



Denominator:



lower bounded by $2\pi\delta 2^m(2/\pi) = 4|\delta|2^m$ for $|\delta| \leq 1/2^{m+1}$.

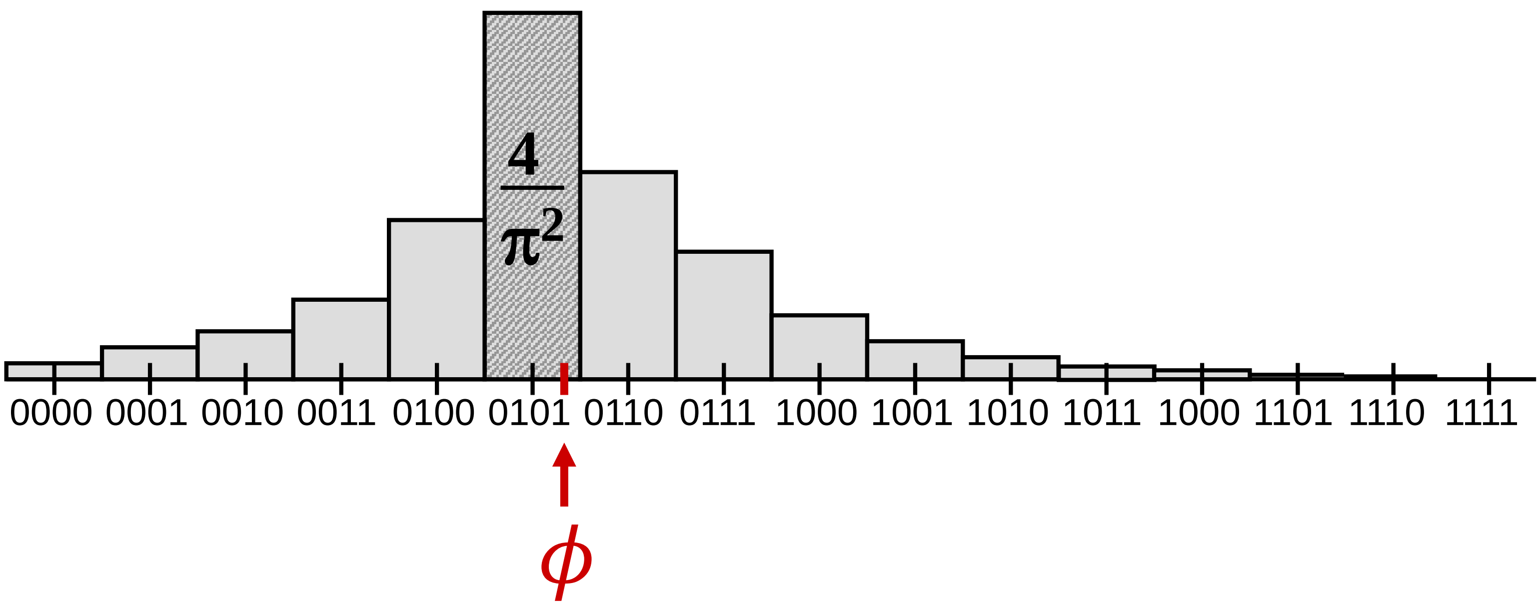
upper bounded by $2\pi\delta$

Therefore, the absolute value of the amplitude of $|y\rangle$ is at least $(1/2^m) \times$ (numerator/denominator) $= 2/\pi$.

Algorithm for eigenvalue estimation (6)

Therefore, the probability of measuring an m-bit approximation of ϕ is always at least $4/\pi^2 \approx 0.4$.

For example, when $\phi = \frac{1}{3} = 0.0101010101010101\ldots$,
the outcome probabilities look roughly like this:



Note: with $\boldsymbol{2m}$**-qubit** control gate, error probability is exponentially small

Thursday class in **QNC 1501**

Oct. 3	OPT 309	Oct. 5	**QNC 1501**
Oct. 10	QNC 0101	Oct. 12	QNC 0101
Oct. 17	QNC 0101	Oct. 19	QNC 0101
Oct. 24	QNC 0101	Oct. 26	**QNC 1501**
Oct. 31	QNC 0101	Nov. 2	QNC 0101
Nov. 7	QNC 0101	Nov. 9	QNC 0101
Nov. 14	QNC 0101	Nov. 16	QNC 0101
Nov. 21	QNC 0101	Nov. 23	QNC 0101
Nov. 28	QNC 0101	Nov. 30	QNC 0101