# Quantum Algorithms
# Lecture 26
# Shor's algorithm

Zhejiang University

# Introduction

# Procedure for finding a nontrivial divisor

**Input.** An integer $y$ ($y > 1$).

**Step 1.** Check $y$ for parity. If $y$ is even, then give the answer "2"; otherwise proceed to Step 2.

**Step 2.** Check whether $y$ is the $k$-th power of an integer for $k = 2, \ldots, \log_2 y$. If $y = m^k$, then give the answer "$m$"; otherwise proceed to Step 3.

**Step 3.** Choose an integer $a$ randomly and uniformly between 1 and $y - 1$. Compute $b = \gcd(a, \boldsymbol{y})$ (say, by Euclid's algorithm). If $b > 1$, then give the answer "$b$"; otherwise proceed to Step 4.

# Procedure for finding a nontrivial divisor

**Step 4.** Compute $r = per_y(a)$ (using the period finding algorithm that we assume we have). If $r$ is odd, then the answer is "$y$ is prime" (which means that we give up finding a nontrivial divisor). Otherwise proceed to Step 5.

**Step 5.** Compute $d = \gcd(a^{r/2} - 1, y)$. If $d > 1$, then the answer is "$d$"; otherwise the answer is "$y$ is prime".

# Procedure for finding a nontrivial divisor

**Step 4.** Compute $r = per_y(a)$ .

**Step 5.** Compute $d = \gcd(a^{r/2} - 1, y)$. If $d > 1$, then the answer is "$d$"; otherwise the answer is "$y$ is prime".

For example, if $y = 21$ and:

- if $a = 2$, algorithm will find $d = 7$: $r = 6$, because $2^6 = 64 = 1(mod\ 21)$; $\gcd(2^3 - 1,21) = 7$.
- if $a = 5$ will fail to find $d > 1$: $r = 6$, because $5^6 = 64 = 1(mod\ 21)$; $\gcd(5^3 - 1, y) = \gcd(124,21) = 1$

# Shor's algorithm summary

- Pick $a$ randomly in the range 1 to $y-1$, such that $\gcd(a, y) = 1$.
- Use order (period) finding algorithm to find order of a $(mod\ y)$, which will be denoted by $r$.
- If $r$ is even, and $a^{r/2} \neq -1\ (mod\ y)$, then compute $gcd(a^{r/2}$–1,y) and $gcd(a^{r/2}$+1,y).
- Test to see if one of these is a non-trivial factor. If so return, otherwise the algorithm fails. If that is the case, repeat.

# Fourier Transform

# Discrete Fourier Transform

Input is $N$-dimensional complex vector.

$$DFT\begin{pmatrix} x_0 \\ x_1 \\ \dots \\ x_{N-1} \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \dots \\ y_{N-1} \end{pmatrix}$$

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{\frac{2\pi i j k}{N}} x_j$$

# Discrete Fourier Transform

In this example $N = 2$, $x_0 = 1$ and $x_1 = 2$. Hence, we have

$$y_k = \frac{1}{\sqrt{2}} \sum_{j=0}^{1} e^{\frac{2\pi i jk}{2}} x_j.$$

Replacing $k = 0$,

$$y_0 = \frac{1}{\sqrt{2}} \sum_{j=0}^{1} e^{\frac{2\pi i j \cdot 0}{2}} x_j = \frac{1}{\sqrt{2}}(x_0 + x_1) = \frac{3}{\sqrt{2}}$$

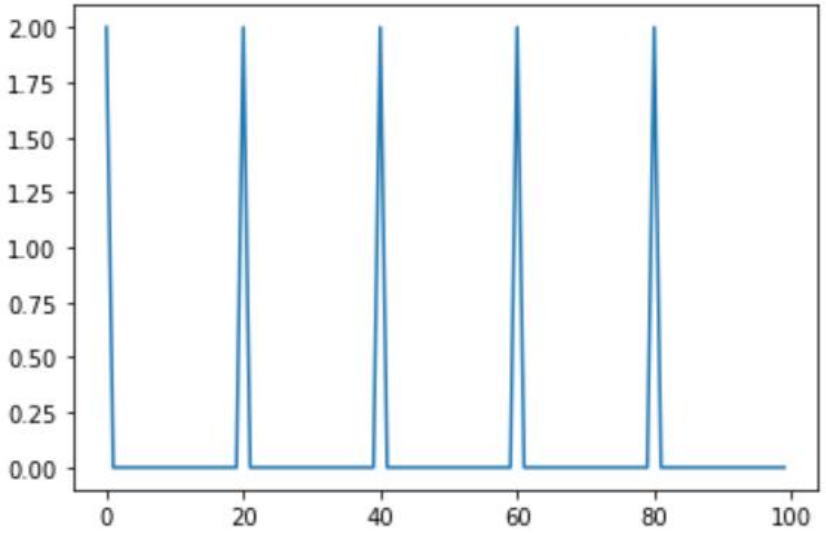
and $k = 1$,

$$y_1 = \frac{1}{\sqrt{2}} \sum_{j=0}^{1} e^{\frac{2\pi i j \cdot 1}{2}} x_j = \frac{1}{\sqrt{2}} \left( e^{\frac{2\pi i \cdot 0 \cdot 1}{2}} x_0 + e^{\frac{2\pi i \cdot 1 \cdot 1}{2}} x_1 \right) = \frac{1 + 2e^{\pi i}}{\sqrt{2}} = \frac{-1}{\sqrt{2}}.$$

We can conclude that $y = \begin{pmatrix} \frac{3}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix}$.

# Discrete Fourier Transform

If we are given a vector of 100 elements where all elements are 0 and each 5-th element is 1 and apply DFT, we get:




When a periodic list of numbers is provided as input to DFT, then the transformed list have peaks around the integer multiples of $N/r$ where $N$ is the number of elements in the list and $r$ is the period.

# QFT

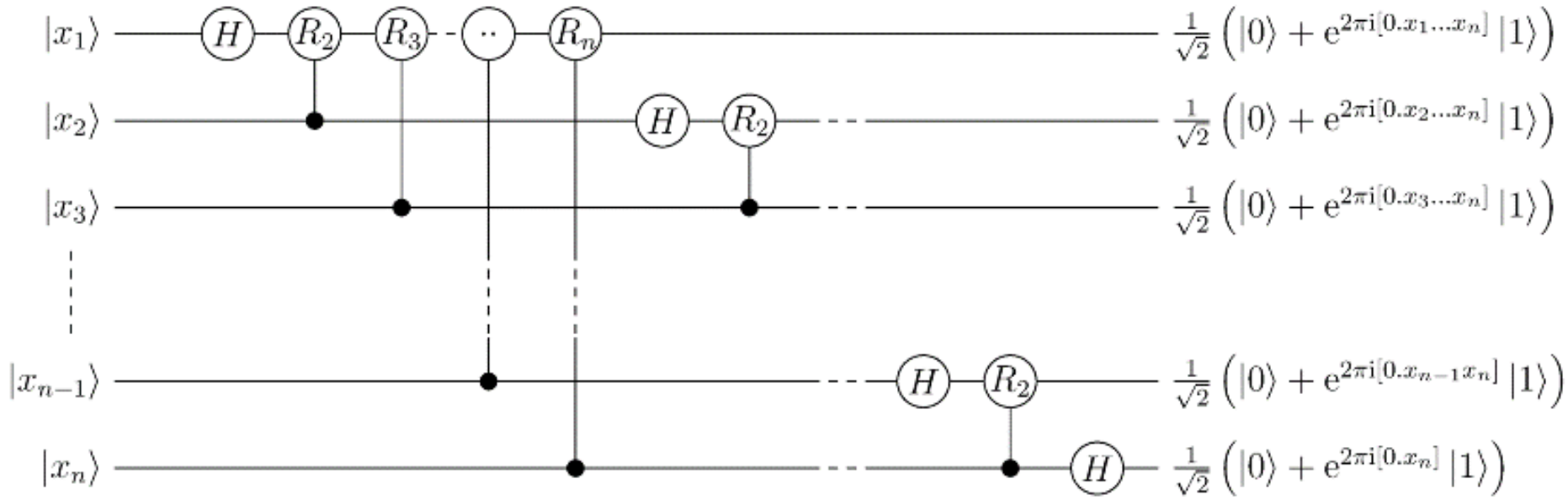
Quantum Fourier Transform - matrix form:

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

QFT can be efficiently used for Period finding. Here $\omega = e^{2\pi i/N}$ is root of unity. The best quantum Fourier transform algorithms known (as of late 2000) require only $O(\mathrm{nlog}\, n)$ gates to achieve an efficient approximation.

$$\mathrm{QFT} : |x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{xk} |k\rangle$$

# QFT




$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad \text{and} \quad R_m = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^m}} \end{pmatrix}$$

Controlled phase gates $R_m$ can be implemented with sequence: $\mathrm{CNOT}\ (-R_{m-1})\ \mathrm{CNOT}\ R_{m-1}$.

# QFT

To implement $QFT^{\dagger}$, one should apply all the operations in reverse order to undo the circuit.

$$QFT^{\dagger}|k\rangle = \frac{1}{\sqrt{N}} \sum_{l=0}^{N-1} e^{-\frac{2\pi i j k}{N}} |l\rangle$$

Implementation from previous slide takes $O(n^2)$ gates.

# Measuring operator

# Measuring operator

$$W = \sum_j \Pi_{L_j} \otimes U_j$$

We have a state space $N \otimes K$, $N = \bigotimes_{j \in \{1,\dots,r\}} L_j$ (pairwise orthogonal subspaces), $\Pi_{L_j}$ is a projection on a subspace $L_j$, $U_j \in L(K)$.

We have projections in space $N$, if a system appears in subspace $L_j$, then $U_j$ is applied to subspace $K$.

# Measuring

We want to measure $\rho \in L(N)$.
First, add subsystem: joint state is $\rho \otimes |0\rangle\langle 0|$.
Then, apply $W = \sum_j \Pi_{L_j} \otimes U_j$.

$$W\left(\rho \otimes |0^m\rangle\langle 0^m|\right)W^\dagger = \sum_j \left(\Pi_{\mathcal{L}_j}\rho\Pi_{\mathcal{L}_j}\right) \otimes \left(U_j|0\rangle\langle 0|U_j^\dagger\right)$$

Make additional space classical by applying the decoherence transformation:

$$U_j|0\rangle\langle 0|U_j^\dagger \mapsto \sum_k \left|\langle k|U_j|0\rangle\right|^2 |k\rangle\langle k|$$

We obtain

$$\sum_j \sum_k \left(\Pi_{\mathcal{L}_j}\rho\Pi_{\mathcal{L}_j}\left|\langle k|U_j|0\rangle\right|^2, k\right) = \sum_j \sum_k \mathbf{P}(k|j) \cdot \left(\Pi_{\mathcal{L}_j}\rho\Pi_{\mathcal{L}_j}, k\right)$$

# Example

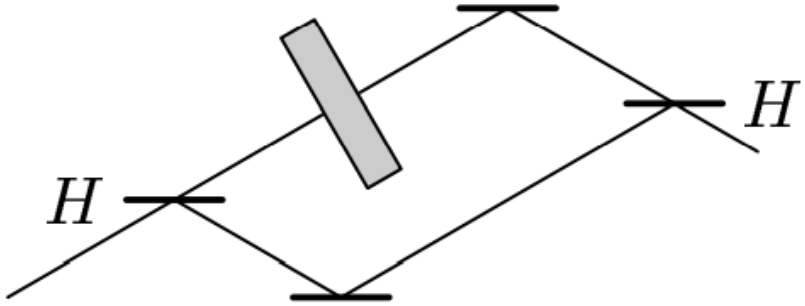
The first example is just a unitary operation, controlled by one qubit: $\Lambda(U) = \Pi_0 \otimes I + \Pi_1 \otimes U$.

Here, we apply $U$ to the second subsystem, if first qubit is in state 1.

# Example

$$\Xi(U) = (H \otimes I)\Lambda(U)(H \otimes I)\colon B^{\otimes N} \to B^{\otimes N}$$

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




If the initial vector has the form $|\psi\rangle = |\eta\rangle \otimes |\xi\rangle$ ($|\xi\rangle \in L_j$), then $\Xi(U)|\psi\rangle = |\eta'\rangle \otimes |\xi\rangle$, where

$$|\eta'\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & \lambda_j \end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}|\eta\rangle = \frac{1}{2}\begin{pmatrix} 1+\lambda_j & 1-\lambda_j \\ 1-\lambda_j & 1+\lambda_j \end{pmatrix}|\eta\rangle$$

# Example

$$|\eta'\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & \lambda_j\end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}|\eta\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda_j & 1-\lambda_j\\ 1-\lambda_j & 1+\lambda_j\end{pmatrix}|\eta\rangle$$

Here $\lambda_j$ is a phase shift applied to the qubit $|\eta'\rangle$, so that amplitude of state $|1\rangle$ is multiplied by $\lambda_j = e^{\pi i \varphi_j}$.

$$\Xi(U) = \sum_j \overbrace{\frac{1}{2}\begin{pmatrix}1+\lambda_j & 1-\lambda_j\\ 1-\lambda_j & 1+\lambda_j\end{pmatrix}}^{R_j} \otimes \Pi_{\mathcal{L}_j}$$

We obtain the following conditional probabilities:

$$\mathbf{P}(0|j) = \left|\langle 0|R_j|0\rangle\right|^2 = \left|\frac{1+\lambda_j}{2}\right|^2 = \frac{1+\cos(2\pi\varphi)}{2}$$

# Example

$$|\eta'\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & \lambda_j\end{pmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}|\eta\rangle = \frac{1}{2}\begin{pmatrix}1+\lambda_j & 1-\lambda_j\\ 1-\lambda_j & 1+\lambda_j\end{pmatrix}|\eta\rangle$$

```
phase shift f = 0.25
cos(2pi f) = 6.123233995736766e-17
probability of state |0> = [1+cos(2pi f)]/2 = 0.5
{'1': 5010, '0': 4990}
```


$$\mathbf{P}(0|j) = \left|\langle 0|R_j|0\rangle\right|^2 = \left|\frac{1+\lambda_j}{2}\right|^2 = \frac{1+\cos(2\pi\varphi)}{2}$$

# Operator for measuring eigenvalues

$$|0\rangle \otimes |\xi_k\rangle \xrightarrow{\Xi(U_a)} \left( \frac{1 + e^{2\pi i \varphi_k}}{2} |0\rangle + \frac{1 - e^{2\pi i \varphi_k}}{2} |1\rangle \right) \otimes |\xi_k\rangle$$

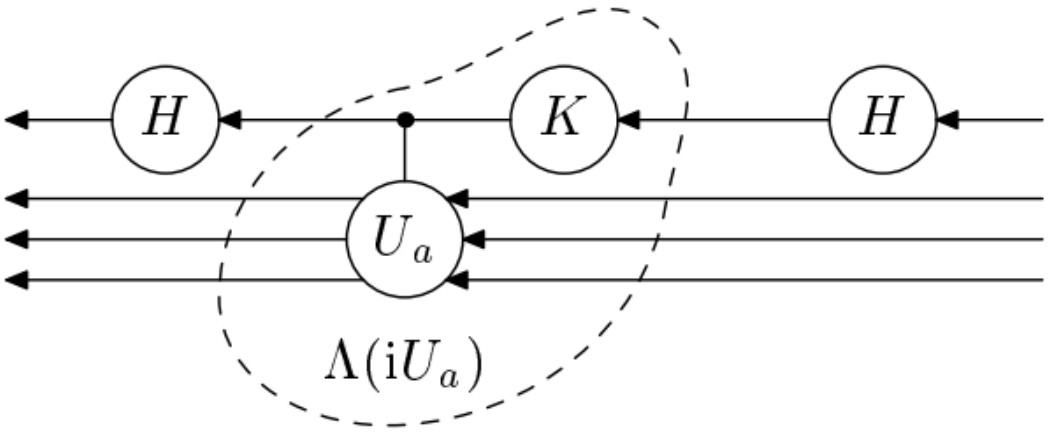
We get conditional probabilities for the first qubit:

$$\mathbf{P}(0|k) = \left| \frac{1 + e^{2\pi i \varphi_k}}{2} \right|^2 = \frac{1 + \cos(2\pi\varphi_k)}{2}, \qquad \mathbf{P}(1|k) = \frac{1 - \cos(2\pi\varphi_k)}{2}$$

Although the conditional probabilities depend on $\phi_k$, they do not allow one to distinguish between $\phi_k = \phi$ and $\phi_k = -\phi$ (like in case of global phase). That is why another type of measurement is needed.

# Another operator - improvement

We will use the operator $\Xi(iU_a)$. $K = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ is from the standard basis.



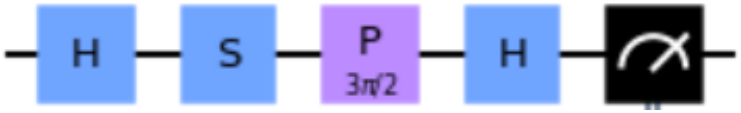

The encircled part of the diagram realizes the operator $\Lambda(iU_a)$. Indeed, $K$ multiplies only $|1\rangle$ by $i$, but this is just the case where the operator $U_a$ is applied (by the definition of $\Lambda(U_a)$).

# Another operator - improvement

For the operator $\Xi(iU_a)$ the conditional probabilities are

$$\mathbf{P}(0|k) = \frac{1-\sin(2\pi\varphi_k)}{2}, \qquad \mathbf{P}(1|k) = \frac{1+\sin(2\pi\varphi_k)}{2}$$

```
phase shift f = 0.75
sin(2pi f) = -1.0
probability of state |0> = [1-sin(2pi f)]/2 = 1.0
{'0': 10000}
```




Here S-gate is our operator $K = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$.

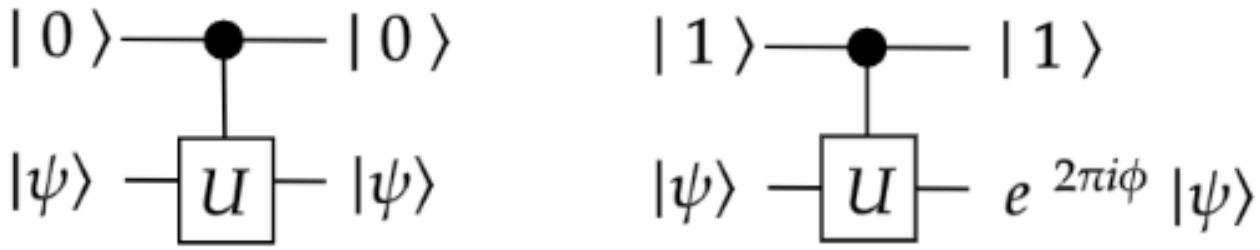
# Phase estimation

# General observations

Let $U$ be a unitary operator with eigenstate $|\psi\rangle$ and eigenvalue $e^{2\pi i\phi}$ such that $U|\psi\rangle = e^{2\pi i\phi}|\psi\rangle$. Let's check the effect of controlled-$U$ ($CU$) operator on a two qubit system where second qubit is set to $|\psi\rangle$.

In general, for any unitary operator $U$ with $U|\psi\rangle = e^{2\pi i\phi}|\psi\rangle$, and its controlled version $CU$ the following is true.

$$CU(|0\rangle|\psi\rangle) \rightarrow |0\rangle|\psi\rangle \quad \text{and} \quad CU(|1\rangle|\psi\rangle) \rightarrow e^{2\pi i\phi}|1\rangle|\psi\rangle$$



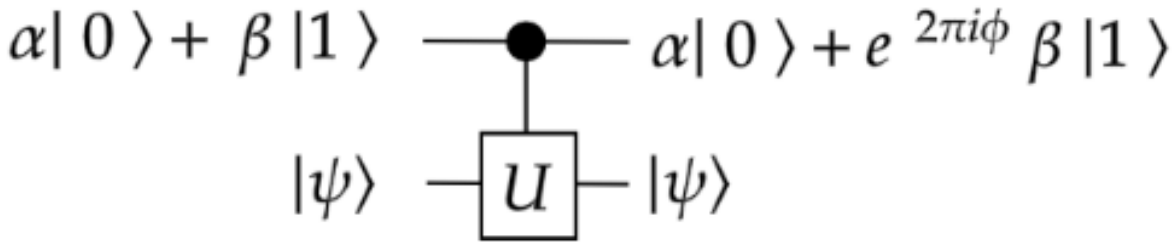
# General observations

$CU$ operator puts a phase of $e^{2\pi i\phi}$ in front of state $|1\rangle$ when the first qubit is in superposition of the states $|0\rangle$ and $|1\rangle$.

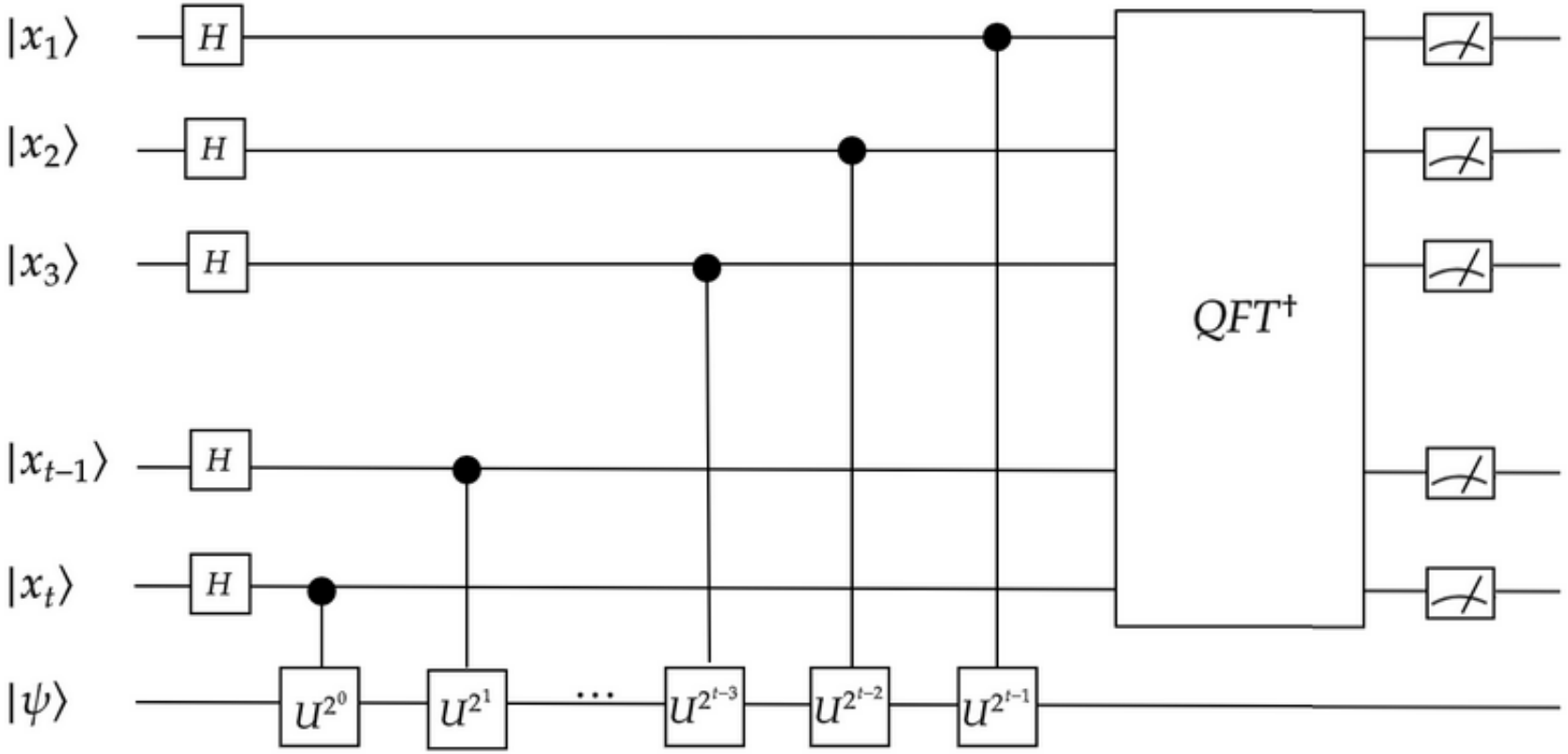
$$CU\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}|\psi\rangle\right)\rightarrow\frac{|0\rangle+e^{2\pi i\phi}|1\rangle}{\sqrt{2}}|\psi\rangle$$

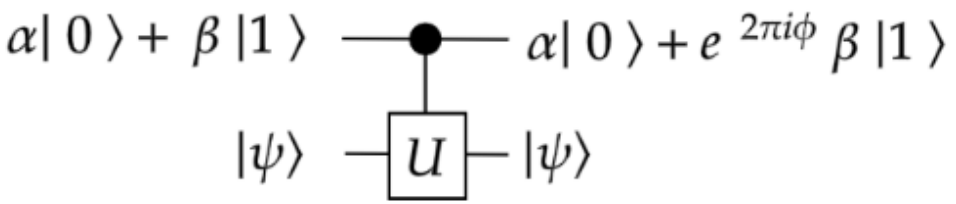
Hence, for an arbitrary state,

$$\alpha|0\rangle|\psi\rangle+\beta|1\rangle|\psi\rangle\xrightarrow{CU}\alpha|0\rangle|\psi\rangle\ +\ e^{2\pi i\phi}\beta|1\rangle|\psi\rangle=(\alpha|0\rangle\ +\ e^{2\pi i\phi}\beta|1\rangle)|\psi\rangle.$$



# The algorithm



# Remark

If you take any arbitrary state instead of $|\psi\rangle$, then you obtain an approximation to one of the eigenvalues with some probability. The reason behind is that you can express any quantum state as the linear combination of the eigenvectors.

# Order finding

# Main idea

For positive integers a and y where a<y with no common factors, order of a is the least positive integer $r$ such that $a^r = 1(mod\ y)$. In order finding algorithm, given a and y, our goal is to find $r$.

Let a<y be given. The idea is to apply phase estimation to the operator $U_a$ which maps $U_a|x\rangle \rightarrow |ax(\mathrm{mod}y)\rangle$ where $\mathrm{x} \in \{0,1\}^L$ and $0 \leq x \leq y-1$. We assume that $U$|x⟩=|x⟩ if y$\leq x \leq 2^L - 1$.

# Eigenvectors

Now let's prove that $|u_s\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{-\frac{2\pi i s k}{r}} |x^k \pmod N\rangle$ for $0 \le s \le r-1$ are eigenvectors for $U_x$ with the corresponding eigenvalues $e^{\frac{2\pi i s}{r}}$.

$$
\begin{aligned}
U_x|u_s\rangle &= \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{\frac{-2\pi i s k}{r}} |x^{k+1} \pmod N\rangle \\
&= \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{\frac{-2\pi i s (k+1)}{r}} e^{\frac{2\pi i s}{r}} |x^{k+1} \pmod N\rangle \quad \text{since } x^r = 1 \pmod N \\
&= e^{\frac{2\pi i s}{r}} |u_s\rangle
\end{aligned}
$$

$$
\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle = |1\rangle
$$

# Procedure

We use two registers: First register has $t$ qubits, second register has $L$ qubits. Let $t = 2L + 1 + \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil$. Choice of $t$ will become clear later on.

- Initialize the registers as

$$|\psi_0\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |0\rangle|u_s\rangle = |0\rangle|1\rangle.$$

Note that here by $|0\rangle$, we denote $|0\rangle^{\otimes t}$ and by $|1\rangle$ we denote $|0\rangle^{L-1}|1\rangle$.

- Apply $H$ and $CU^{2^j}$ gates in the phase estimation algorithm.

$$|\psi_1\rangle = \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} \frac{1}{2^{t/2}} \sum_{k=0}^{2^t-1} e^{\frac{2\pi i s k}{r}} |k\rangle|u_s\rangle$$

# Procedure

- Apply Inverse QFT to the first register.

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |\tilde{\phi}\rangle |u_s\rangle$$

At the end of this procedure, for each $s$ in the range $0, \ldots, r-1$, we obtain an estimate of the phase $\tilde{\phi} = \frac{s}{r}$ accurate to $2L+1$ bits with probability at least $\frac{1-\epsilon}{r}$.

Note that if $r$ is not a power of 2, then it can not be expressed in the form $\frac{x}{N}$ for some $x$ and $N = 2^t$.

Now the question is how to find $r$ from the estimate of $s/r$? The answer is using continued fractions.

# Continued fractions

The second part of Theorem A.13 guarantees that the number $s/r$ is contained among the convergents.

$$2 + \cfrac{1}{3 + \cfrac{1}{1 + \frac{1}{2}}}$$

Find the continued fraction for $3.245 = \frac{649}{200}$

| Step | Real Number | Integer part | Fractional part | Simplified | Reciprocal of $f$ |
|---|---|---|---|---|---|
| 1 | $r = \frac{649}{200}$ | $i = 3$ | $f = \frac{649}{200} - 3$ | $= \frac{49}{200}$ | $\frac{1}{f} = \frac{200}{49}$ |
| 2 | $r = \frac{200}{49}$ | $i = 4$ | $f = \frac{200}{49} - 4$ | $= \frac{4}{49}$ | $\frac{1}{f} = \frac{49}{4}$ |
| 3 | $r = \frac{49}{4}$ | $i = 12$ | $f = \frac{49}{4} - 12$ | $= \frac{1}{4}$ | $\frac{1}{f} = \frac{4}{1}$ |
| 4 | $r = 4$ | $i = 4$ | $f = 4 - 4$ | $= 0$ | STOP |

Continued fraction form for $3.245 = \frac{649}{200} = [3; 4, 12, 4]$

$$= 3 + \cfrac{1}{4 + \cfrac{1}{12 + \frac{1}{4}}}$$

The convergents are $c_1 = 2$, $c_2 = 2 + \frac{1}{3} = \frac{7}{3}$, $c_3 = 2 + \cfrac{1}{3 + \frac{1}{1}} = \frac{9}{4}$, $c_4 = 2 + \cfrac{1}{3 + \cfrac{1}{1 + \frac{1}{2}}} = \frac{25}{11}$

# Remaining parts

Modular exponentiation can be performed efficiently to apply $CU^{2^j}$ (lecture 24, slides 23-24).

We repeat algorithm for several times to obtain different irreducible fractions $s'/r'$, to find initial period $r$ we will need to find least common denominator for our fractions.

# Complexity summary

Overall, we have an algorithm which uses $O(L^3)$ gates, $O(L)$ qubits and constant repetitions.

- Hadamard operation at the beginning requires $O(L)$ gates;
- $O(L^2)$ gates are required by $QFT^\dagger$
- $O(L^3)$ gates are needed for modular exponentiation
- Continued fraction algorithm requires $O(L^3)$ classical processing

Here $L = logN$

# Shor’s algorithm

# Sources for implementation

This page contains description of implementation of Shor's algorithm with quantum programming:
https://qiskit.org/textbook/ch-algorithms/shor.html

# Problem definition

Given a positive integer $y$, what prime numbers when multiplied together equal to $y$?

It should also be noted that the converse is an easy problem. If the prime numbers are already given, they can be multiplied to check whether they are really the factors of $y$. This property of the problem lies at the center of encryption algorithms which is widely used today.

# Source of efficiency

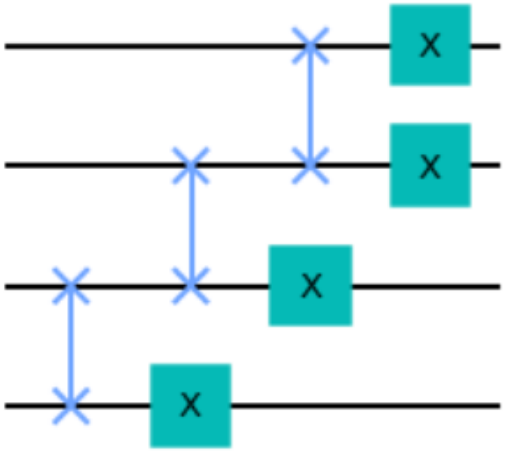
Factorization problem can be reduced to order finding problem.

At the background of Shor's algorithm lies the exponential speed up that comes from Quantum Fourier Transform.

# Order finding – main technique

Shor's algorithm uses order finding algorithm which is a special case of quantum phase estimation where the operator whose phase is estimated is $U_a|x\rangle \rightarrow |ax\ (mod\ y)\rangle$.

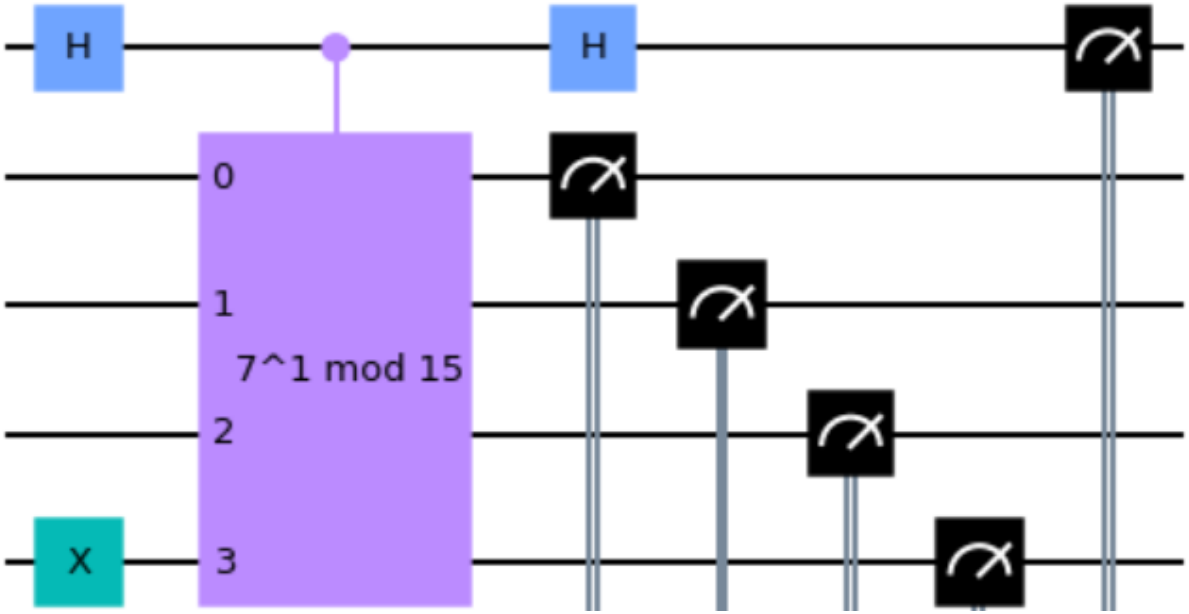
Example for $y = 15$, $a = 7$:



# Measuring operator example

Measuring operator for $U_7|x\rangle \rightarrow |7x\ (mod\ 15)\rangle$.

Eigenstate for this operator is $|1\rangle$, first qubit will keep outcome of measurement, other 4 qubits are needed to apply operator to the eigenstate.



# States in order finding

- Start with the state $|0\rangle|1\rangle$.
- Apply Hadamard to first register.

$$\frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} |k\rangle|1\rangle$$

- Apply controlled operations.

$$\frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} |k\rangle|x^k \pmod{N}\rangle$$

# States in order finding

- Measure the second register and continue with the first register.

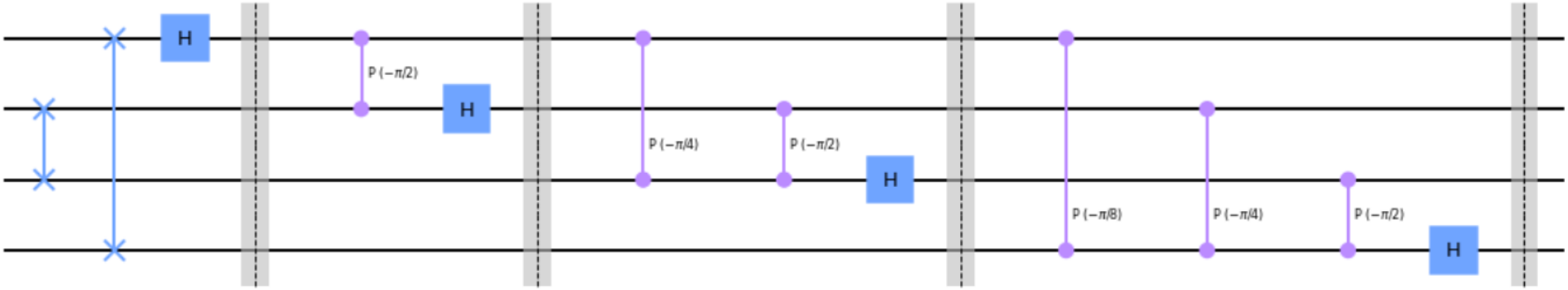
$$\sqrt{\frac{r}{2^t}} \sum_{x=0}^{2^t/r-1} |x_0 + xr\rangle$$

- Apply inverse $QFT$ to the first register.

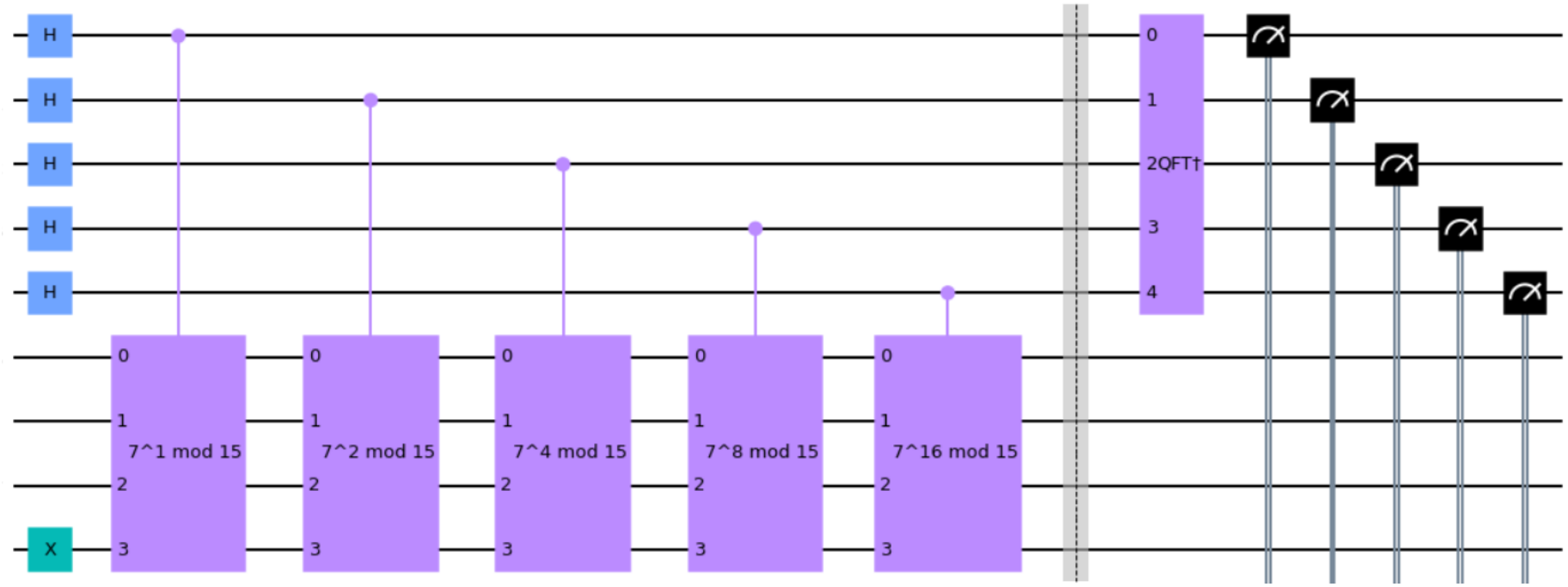
$$\sqrt{\frac{r}{2^t}} \sum_{k=0}^{2^t-1} \frac{1}{\sqrt{2^t}} \sum_{x=0}^{2^t/r-1} e^{-\frac{2\pi i(rx+x_0)k}{2^t}} |k\rangle$$

- Measure the first register. Divide the state you observe by $2^t$ and apply continued fractions algorithm.

# Inverse QFT



# Phase estimation



# Outcomes

| | Register Output | Phase |
|---|---|---|
| 0 | 11000(bin) = 24(dec) | 24/32 = 0.75 |
| 1 | 10000(bin) = 16(dec) | 16/32 = 0.50 |
| 2 | 01000(bin) = 8(dec) | 8/32 = 0.25 |
| 3 | 00000(bin) = 0(dec) | 0/32 = 0.00 |

| | Phase | Fraction | Guess for r |
|---|---|---|---|
| 0 | 0.75 | 3/4 | 4 |
| 1 | 0.50 | 1/2 | 2 |
| 2 | 0.25 | 1/4 | 4 |
| 3 | 0.00 | 0/1 | 1 |

$$\gcd\left(7^{\frac{r}{2}} - 1, 15\right) = \gcd(7^2 - 1, 15) = 3$$

# Outcomes

```
Attempt 1:
Register Reading: 00000000
Corresponding Phase: 0.000000
Result: r = 1

Attempt 2:
Register Reading: 10000000
Corresponding Phase: 0.500000
Result: r = 2
Guessed Factors: 3 and 1
*** Non-trivial factor found: 3 ***
```

# Thank you for your attention!