Quantum Fourier Transform (QFT)

◈ Definition

The Quantum Fourier Transform (QFT) is the quantum version of the Discrete Fourier Transform (DFT) used in classical signal processing.
It transforms a quantum state from the computational basis (e.g., |0⟩, |1⟩, …) into the frequency domain, revealing periodic patterns hidden in amplitudes.
In simpler words:
QFT helps a quantum computer find periodicity (r) — which is the key to Shor's Algorithm.

Mathematical Idea
For an n-qubit quantum system, we can represent the state as:

$$| x\rangle = | x_{n-1}x_{n-2}\dots x_0\rangle$$

The Quantum Fourier Transform acts as:

$$QFT \mid x\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2\pi i x k/2^n} \mid k\rangle$$

Shor's Algorithm (Factoring & Cryptography)

◈ What is it?

Shor's Algorithm, developed by *Peter Shor in 1994*, is a **quantum algorithm** designed to factor large integers efficiently — something that **classical computers struggle with**.
This algorithm is important because it can **break RSA encryption**, which relies on the difficulty of factoring large numbers.

Why is factoring important?
Let's say you have a number:

$$N = p \times q$$

where $p$and $q$are prime numbers.

- In classical computers, finding $p$and $q$from $N$is extremely hard when $N$is large (hundreds or thousands of bits).
- In quantum computers, **Shor's Algorithm** can do this **exponentially faster**.

This threatens cryptographic systems like **RSA**, which depend on the hardness of factoring.

Mathematical Foundation
The algorithm finds the **period (r)** of the function:

$$f(x) = a^x \bmod N$$

where:

- N= number to factor
- a= random integer such that $1 < a < N$and $\gcd(a, N) = 1$

If the **period** $r$satisfies:

$$a^r \equiv 1 \pmod{N}$$

then the **factors of** N can be found by computing:

$$\text{factors} = \gcd(a^{r/2} - 1, N) \text{and} \gcd(a^{r/2} + 1, N)$$

How Shor's Algorithm Works (Step-by-Step)

Step	Type	Description
1	Classical	Pick a random number a such that $\gcd(a, N) = 1$.
2	Quantum	Create a superposition of all possible exponents x.
3	Quantum	Compute $f(x) = a^x \bmod N$ using **modular exponentiation**.
4	Quantum	Apply the **Quantum Fourier Transform (QFT)** to find the period r.
5	Classical	Use the value of r to calculate the factors of N.
6	Classical	Verify that the found numbers multiply to N.

Quantum Fourier Transform (QFT)

The **QFT** is the **quantum version of the Discrete Fourier Transform (DFT)**.

It transforms the quantum state so that the **period of f(x)** appears as peaks in the probability amplitudes of measurement outcomes.

It's the most crucial quantum step — it **extracts the hidden periodicity** from the modular exponentiation function.

Example: Factoring 15

Let's factor $N = 15$:

1. Choose $a = 7$ (coprime with 15).
2. zzzCompute $f(x) = 7^x \bmod 15$:
   $7^1 = 7, 7^2 = 4, 7^3 = 13, 7^4 = 1$
   → Period $r = 4$
3. Compute:
   $$\gcd(7^{4/2} - 1, 15) = \gcd(48, 15) = 3$$
   $$\gcd(7^{4/2} + 1, 15) = \gcd(50, 15) = 5$$
4. Factors = 3 and 5

Why Shor's Algorithm is Powerful

Property	Classical	Quantum (Shor)
Time Complexity	$O(e^{(\log N)^{1/3}})$	$O((\log N)^3)$
Scaling	Exponential	Polynomial
Practical Impact	Secure RSA	Breaks RSA

Cryptography Implications

RSA encryption uses:

$$N = p \times q$$

and relies on the fact that factoring $N$is hard.
If large-scale quantum computers become practical, **RSA, Diffie–Hellman, and ECC** will become insecure.
Hence, research is ongoing into **Post-Quantum Cryptography (PQC)** — cryptographic methods resistant to quantum attacks.

Shor's algorithm — detailed, step-by-step math (deep dive)
Great — here's a thorough, math-heavy, step-by-step explanation of **Shor's algorithm**, including the number-theory facts, the quantum-period-finding derivation, how to recover the period using **continued fractions**, success probability, resource counts, and a worked numeric example.

---

1 — Goal and high-level structure
Goal: Given a composite integer N, find a nontrivial factor p(so $1 < p < N$and $p \mid N$).
High-level strategy. Reduce factoring to **period finding**:

1. Choose random integer $a$with $1 < a < N$.
2. If $\gcd(a, N) > 1$then we already found a factor.
3. Otherwise, find the *order* $r$of $a$modulo N:

$$r = \min\{r' > 0 \mid a^{r'} \equiv 1 (\mathrm{mod} N)\}.$$

4. If $r$is even and $a^{r/2} \not\equiv -1 (\mathrm{mod} N)$, then

$$p = \gcd(a^{r/2} - 1, N), q = \gcd(a^{r/2} + 1, N)$$

give nontrivial factors (with high probability).

The **quantum** part is used to find $r$efficiently (period finding). The rest is classical arithmetic.

---

2 — Why order $r$gives factors

Assume $\gcd(a, N) = 1$and $r$is the order. Then $a^r \equiv 1 (\mathrm{mod} N)$.

If $r$is even, write $a^r - 1 = (a^{r/2} - 1)(a^{r/2} + 1)$. Since $a^r \equiv 1$, both terms are multiples of factors of N. If neither factor is congruent to 0 mod N(i.e., neither equals $0 mod N$), then $\gcd(a^{r/2} \pm 1, N)$yields a nontrivial factor. The failure case is when $a^{r/2} \equiv -1 (\mathrm{mod} N)$(then both gcds give 1 or N). Randomly choosing $a$makes such failures uncommon; repeating the procedure few times suffices.

3 — The quantum period-finding subroutine (precise math)

Let $n = \lceil \log_2 N \rceil$. Choose integer Qas a power of 2 with $N^2 \leq Q < 2N^2$(commonly $Q = 2^m$for some m). The quantum circuit uses two registers:

- First register: $m$qubits, able to represent integers $0, \ldots, Q-1$.
- Second register: enough qubits to represent integers modulo N($\approx n$qubits).

Initial state

$$\frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} \quad |\, x\rangle \,|\, 0\rangle.$$

Apply modular exponentiation (unitary U_f):

$$U_f : |\, x\rangle \,|\, 0\rangle \mapsto |\, x\rangle \,|\, a^x \bmod N\rangle.$$

After this:

$$|\psi\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |a^x \bmod N\rangle.$$

Measure the second register. Suppose measurement yields the value $v = a^{x_0} \bmod N$. Because $f(x)$is periodic with period r, the first register collapses to a uniform superposition over those $x$congruent to $x_0 (\mathrm{mod} r)$:

$$|\, \psi_{x_0}\rangle = \frac{1}{\sqrt{K}} \sum_{m=0}^{K-1} \quad |\, x_0 + mr\rangle$$

where $K \approx Q/r$(more precisely, $K = \lfloor (Q - x_0 - 1)/r \rfloor + 1$).

Apply Quantum Fourier Transform (QFT) on the first register. QFT on Qelements acts as

$$\mathrm{QFT}_Q : |\, x\rangle \mapsto \frac{1}{\sqrt{Q}} \sum_{s=0}^{Q-1} \quad e^{2\pi i x s/Q} \,|\, s\rangle.$$

Thus QFT applied to $|\, \psi_{x_0}\rangle$gives

$$\mathrm{QFT}_Q \,|\, \psi_{x_0}\rangle = \frac{1}{\sqrt{K}} \cdot \frac{1}{\sqrt{Q}} \sum_{s=0}^{Q-1} \Big(\sum_{m=0}^{K-1} \quad e^{2\pi i (x_0 + mr) s/Q} \Big) \,|\, s\rangle.$$

The inner sum is a geometric series:

$$S(s) = e^{2\pi i x_0 s/Q} \cdot \frac{1 - e^{2\pi i K r s/Q}}{1 - e^{2\pi i r s/Q}}.$$

Its magnitude is large when the denominator is small — i.e., when $rs/Q$is close to an integer. So measurement outcomes $s$are likely to be near integer multiples of Q/r. More precisely, the probability of measuring $s$is roughly

$$\Pr(s) \approx \frac{1}{r} \left| \frac{\sin(\pi K r s/Q)}{\sin(\pi r s/Q)} \right|^2,$$

which concentrates around $s \approx kQ/r$for integers k.

Thus a measurement yields an s such that

$$|\frac{s}{Q}-\frac{k}{r}|\leq\frac{1}{2Q}$$

for some integer $k$with reasonably high probability (we'll quantify success later). From measured s/Qwe can recover the rational k/r(with denominator $r$bounded by N) using the continued fraction algorithm.

---

4 — Continued fractions: recover $r$from measured s

You measure s. Compute the rational approximation s/Q. Use the continued fraction expansion of $s/Q$to find the best rational approximations $p/q$with denominator $q$small ($\leq N$). Among the convergents p/q, search for a candidate denominator $q$that satisfies

$$a^q \equiv 1(\mathrm{mod}N).$$

If found, $q$is the order r(or a divisor of r; if it divides $r$adjust). If the continued-fraction candidate yields a multiple of the order, additional tests can be used.

Worked continued-fraction example (concrete):

Take $N = 15$, $a = 7$, true order $r = 4$. Choose $Q = 256$(a power of two, $Q \geq N^2 = 225$). Possible good $s$values are $s \approx kQ/r$. For $k = 1$: $s \approx 256/4 = 64$. Suppose measured $s = 64$. Then

$$\frac{s}{Q}=\frac{64}{256}=\frac{1}{4}.$$

The continued fraction of $1/4$is $[0;4]$i.e. convergents: $0/1,1/4$. Candidate $q = 4$. Check $7^4 mod 15 = 1$. So $r = 4$recovered.

If the measurement gave $s = 65$, then $65/256 \approx 0.25390625$. Continued fraction steps:

- — compute convergents sequentially until denominator $\leq N$. One convergent will be $1/4$or some approximation to it; test denominators for the order condition.

Continued fractions are guaranteed to find a good rational $k/r$when $|\ s/Q - k/r\ | < 1/(2Q)$and $r < Q$.

5 — Classical checks (after finding candidate r)

Once a candidate $r$is found:

1. If $r$is odd, discard (choose new $a$and repeat).
2. If $r$is even, compute $x = a^{r/2}(\mathrm{mod}N)$.
    - If $x \equiv -1(\mathrm{mod}N)$, failure (repeat with another a).
    - Else compute $p = \gcd(x-1, N)$, $q = \gcd(x+1, N)$. These are factors (usually nontrivial).

Probability that a randomly chosen $a$leads to a successful $r$is at least 1/2 for many N; repeating O(1) times makes success overwhelmingly likely.

6 — Probability of success and repetitions

Two kinds of failures:

- The chosen $a$has $\gcd(a, N) > 1$(actually a success — you found factor).

- The order $r$is odd or $a^{r/2} \equiv -1(\mathrm{mod}N)$. For composite Nthat are not powers, the fraction of $a$that lead to useful even $r$is high (≥ 1/2 for worst-case typical RSA-modulus). Thus only O(1) random choices of $a$are needed in expectation.

From the QFT measurement, you may get an $s$that is not close enough to any k/ror the continued-fraction step might not produce r. That also only requires repeating the quantum part O(1) times to get good $s$with high probability.

7 — Quantum resources and complexity

Let $n = \lceil \log_2 N \rceil$.

- **Number of qubits**: Roughly $2n + O(1)$(first register $m$qubits with $m \approx 2n$because Qchosen so that $Q \approx N^2$; second register $n$qubits to hold modular values; plus ancilla qubits for arithmetic). More careful counts for error-corrected implementations put the logical qubit count at several thousand for breaking RSA-2048.
- **Gate complexity**:
  - Modular exponentiation dominates and uses repeated modular multiplications. Modular multiplication can be implemented in $O(n^2)$or $O(n \log n)$time with advanced arithmetic; naive implementations cost $O(n^2)$elementary operations.
  - Total gate count for modular exponentiation is $O(n^3)$in many constructions (because you must do O(n) multiplications each costing O(n^2)). QFT costs $O(m^2)$gates but is smaller: $O(n^2)$.
- **Overall time complexity** (number of elementary gates): polynomial — often stated as $\tilde{O}(n^3)$(polynomial in $\log N$), exponentially faster than best-known classical factoring algorithms.

8 — Explicit worked example — full flow for $N = 15$

1. **Pick** a: choose $a = 7$. $\gcd(7,15) = 1$.
2. **Order**: compute $7^1 \equiv 7$, $7^2 \equiv 4$, $7^3 \equiv 13$, $7^4 \equiv 1$. So $r = 4$.
3. **Quantum part** (sketch): choose $Q = 256$. Prepare uniform superposition $\frac{1}{\sqrt{256}} \sum_{x=0}^{255} \quad |x\rangle$

Apply controlled modular exponentiation to get $\frac{1}{\sqrt{256}} \sum_x \quad |x\rangle|$ $7^x mod 15\rangle Measure\ second\ register\ to\ collapse\ first\ to\ \sum_m \quad |x_0 +$ $m4\rangle. QFT\ on\ first\ register: measurement\ likely\ yields\ snear\ \frac{kQ}{4} = 64k. Suppose\ s =$ $64. Compute\ continued\ fraction\ of\ 64/256 = 1/$ $4\ to\ get\ candidate\ denominator\ 4. Check\ 7^4 mod 15 = 1. Compute\ 7^2 mod 15 =$ $4. Then\ gcd(4-1,15) = 3, gcd(4+1,15) = 5.$

9 — Concrete continued-fraction steps (algorithmic)

Given rational $y = s/Q$:

1. Compute the continued fraction expansion of y, obtain convergents p_i/q_i.
2. For each convergent $p_i/q_i$with $q_i \leq N$, test whether $a^{q_i} \equiv 1(\mathrm{mod}N)$.
3. If yes, set $r = q_i$(if not, continue).
4. If no convergent works, repeat the quantum run to get a different s.

The continued fraction expansion is obtained via repeated Euclidean division; computing convergents is standard and efficient (polynomial time).

10 — Modular exponentiation (quantum implementation sketch)

We must implement the unitary U_f mapping $|x\rangle |y\rangle \mapsto |x\rangle |y \cdot a^x \bmod N\rangle$. Practical construction uses repeated squaring:

- Precompute $a^{2^0} \bmod N, a^{2^1} \bmod N, a^{2^2} \bmod N, \ldots$.
- For each bit x_j of x (LSB to MSB), conditionally multiply the second register by a^{2^j} when $x_j = 1$. These conditional multiplies are reversible circuits built from controlled modular adders and controlled-swaps, or using advanced adders (Draper adder, Fourier adders) — significant engineering but conceptually straightforward.

Reversibility is ensured by keeping ancillas and uncomputing temporary values. Implementations differ in ancilla cost and gate-depth; optimized designs minimize gate count and depth for fault-tolerant operation.

What is a Bell State?

A **Bell state** is one of the **four maximally entangled two-qubit states** in quantum mechanics.

These states represent the **strongest possible correlations** between two qubits — so strong that measuring one qubit immediately determines the other's state, no matter how far apart they are.

The most common Bell state (and the one we'll prepare) is:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

This means:

- The system is **equally likely** to be in $|00\rangle$ or $|11\rangle$.
- But neither qubit has a definite state individually — only their **joint state** is well-defined.

Intuitive Meaning

Bell states are the **simplest example of quantum entanglement** — the fundamental resource used in:

- **Quantum teleportation**
- **Quantum cryptography (QKD)**
- **Superdense coding**
- **Quantum error correction**
- **Quantum networking**

They show that the two qubits are connected in a way that's impossible in classical systems.

Step-by-Step Bell State Preparation

We start with both qubits initialized in $|0\rangle$:

$$|\psi_0\rangle = |00\rangle$$

Step 1 — Apply a Hadamard gate (H) to the first qubit

The **Hadamard gate** creates a superposition:

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

So now the combined system becomes:

$$|\psi_1\rangle = (H \otimes I)\,|00\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$$

At this point:

- The **first qubit** is in superposition.
- The **second qubit** is still $|0\rangle$.
- The two qubits are *not yet entangled*.

Step 2 — Apply a CNOT gate

The **CNOT (Controlled-NOT)** gate flips the **target qubit** (qubit 2) *only if* the **control qubit** (qubit 1) is $|1\rangle$.

Mathematically:

- $CNOT\,|00\rangle = |00\rangle$
- $CNOT\,|10\rangle = |11\rangle$

So applying CNOT to $|\psi_1\rangle$:

$$|\psi_2\rangle = CNOT\,|\psi_1\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

This is the **Bell state** $|\Phi^+\rangle$.

Mathematical Representation

The circuit's operation in matrix form:

1. **Initial State**

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

2. **Hadamard on qubit 1**

$$H \otimes I = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

Multiply it with $|00\rangle$:

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$$

3. ***Apply CNOT***

$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Acting on the previous state gives:

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

*That's our **Bell state**!*

Why It's Entangled

Try to express the state $\frac{1}{\sqrt{2}}(|00\rangle+|11\rangle)$ *as a product of two individual qubit states:*

$$|\psi\rangle = (a\,|0\rangle + b\,|1\rangle) \otimes (c\,|0\rangle + d\,|1\rangle)$$

Expanding gives:

$$ac\,|00\rangle + ad\,|01\rangle + bc\,|10\rangle + bd\,|11\rangle$$

To match our Bell state:

- $ac = \frac{1}{\sqrt{2}}$
- $bd = \frac{1}{\sqrt{2}}$
- ad=0
- $bc = 0$

No valid a, b, c, dcan satisfy all these at once.
Hence, the Bell state **cannot be separated** — the qubits are **entangled**.

Measurement

When measured:

- *50% chance* $\rightarrow |00\rangle$
- *50% chance* $\rightarrow |11\rangle$
- *0% chance* $\rightarrow |01\rangle or |10\rangle$

*Thus, both qubits always yield the **same result**, even though individually each is random.*