So far in this course: assumption is that adversary is classical.

How do things change if adversaries are quantum? We won't go into detail but will state main results:

Grover's algorithm: Given black-box access to a function $f : [N] \to \{0,1\}$, Grover's algorithm finds an $x \in [N]$ such that $f(x) = 1$ by making $O(\sqrt{N})$ queries to f.

"Searching an unsorted database of size N in time $O(\sqrt{N})$."

- Classically: Searching an unstructured database of size N requires time $\Omega(N)$ — cannot do better than a linear scan.
- Quantum: Grover's algorithm is tight for unstructured search. Any quantum algorithm for the unstructured search problem requires making $\Omega(\sqrt{N})$ queries (to the function / database).

$\Rightarrow$ Quantum computes provide a quadratic speedup for unstructured search, and more broadly, function inversion.

Implications in cryptography: Consider a one-way function over a 128-bit domain. The task of inverting a one-way function is to find $x \in \{0,1\}^{128}$ such that $f(x) = y$ for some fixed target value f. Exhaustive search would take time $\approx 2^{128}$ on a classical computer, but using Grover's algorithm, can perform in time $\approx \sqrt{2^{128}} = 2^{64}$.

$\Rightarrow$ For symmetric cryptography, need to double key-sizes to maintain same level of security (unless there are new quantum attacks on the underlying construction itself.

$\Rightarrow$ Use AES-256 instead of AES-128 (not a significant change!)

Similar algorithm can be applied to obtain a quantum collision-finding algorithm that runs in time $\sqrt[3]{N}$ where N is the size of the domain (compare to $\sqrt{N}$ for the best classic algorithm)

↳ Instead of using SHA-256, use SHA-384 (not a significant change)

↳ The quantum algorithm require a large amount of space, so not clear that this is a significant threat, but even if it were, using hash functions with 384-bits of output suffices for security

Main takeaway: Symmetric cryptography mostly unaffected by quantum computers ~ generally just require a modest increase in key size

↳ e.g., symmetric encryption, MACs, authenticated encryption

Story more complicated for public-key primitives:

- Simon's algorithm and Shor's algorithm provide polynomial-time algorithms for solving discrete log (in any group with an efficiently-computable group operation) and for factoring
- Both algorithms rely on period finding (and more broadly, on solving the hidden subgroup problem)

Intuition for discrete log algorith (as a period finding problem):

- Let $(g, h = g^\alpha)$ be the discrete log instance in a group of prime order p
- Let $f : \mathbb{Z}_p \times \mathbb{Z}_p \to \mathbb{G}$ be the function

$$f(x,y) = g^x h^{-y}$$

- By construction,

$$f(x+\alpha, y+1) = g^{x+\alpha} h^{-y-1} = g^x h^{-y} g^\alpha h^{-1} = g^x h^{-y} = f(x,y)$$

- Thus, the element $(\alpha, -1)$ is the period of f, so using Shor's algorithm, we can efficient compute $(\alpha, -1)$ from (g,h), which yields the discrete log of h

Thus, if large scale quantum computers come online, we will need new cryptographic assumptions for our public-key primitives

↳ All the algebraic assumptions we have considered so far (e.g., discrete log, factoring) are broken

How realistic is this threat? - Lots of progress in building quantum computers recently by both academia and industry (e.g., see initiatives by Google, IBM, etc.)

- To run Shor's algorithm to factor a 2048-bit RSA modulus, estimated to need a quantum computer with ≈ 10000 logical qubits (analog of a bit in classical computers)
  - ↳ With quantum error correction, this requires ≥ 10 million physical qubits to realize
  - ↳ Today: machines with 10s of physical qubits, so still very far from being able to run Shor's algorithm
- Optimistic estimate: At least 20-30 years away

Should we be concerned? Quantum computers would break existing key-exchange and signature schemes

- Signatures: Future adversaries would be able to forge signatures under today's public keys, so if quantum computers come online, we can switch to and only use post-quantum schemes
- Key-Exchange: Future adversaries can break confidentiality of today's messages (i.e., we lose forward secrecy) — this is problematic in many scenarios (e.g., businesses want trade secrets to remain hidden for 50 years)

This course: will just focus on getting post-quantum signatures (will not discuss post-quantum key exchange)

[General approach for post-quantum cryptography: base hardness on assumptions believed to be hard on quantum computers (e.g., lattice-based cryptography, isogeny-based cryptography)]

For digital signatures, we will show that OWFs $\Rightarrow$ digital signatures

↳ Signatures can be based on symmetric primitives, so gives one approach to post-quantum signatures