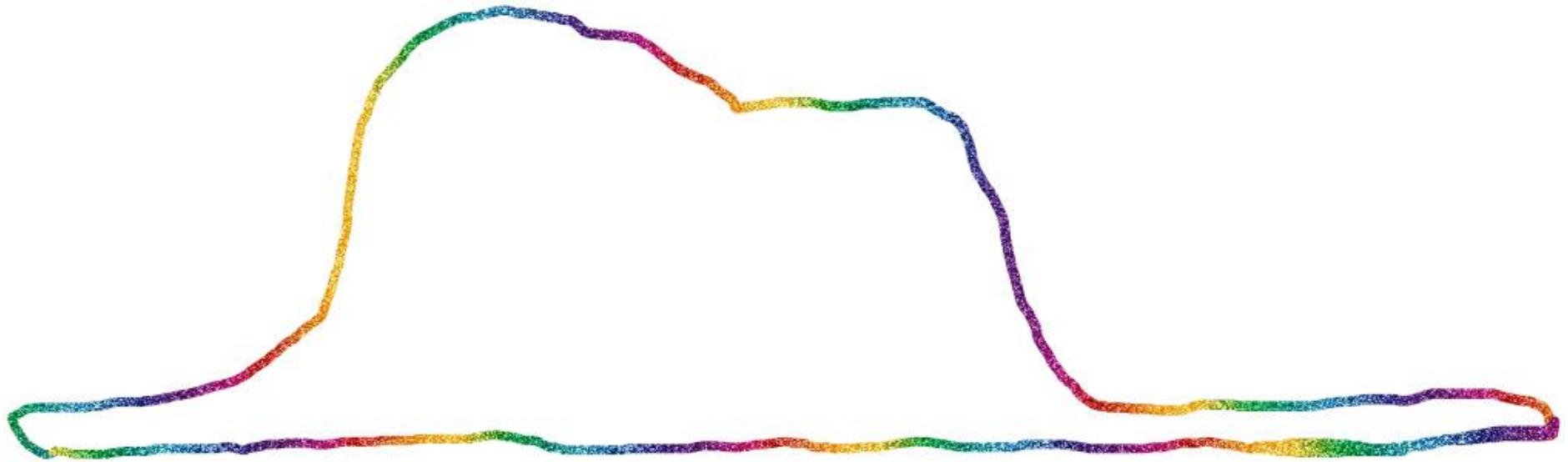
The Hitchhiker's Guide to the Quantum Computing

Joonion Bae
joonion@gmail.com



이 그림은 무슨 그림일까요?

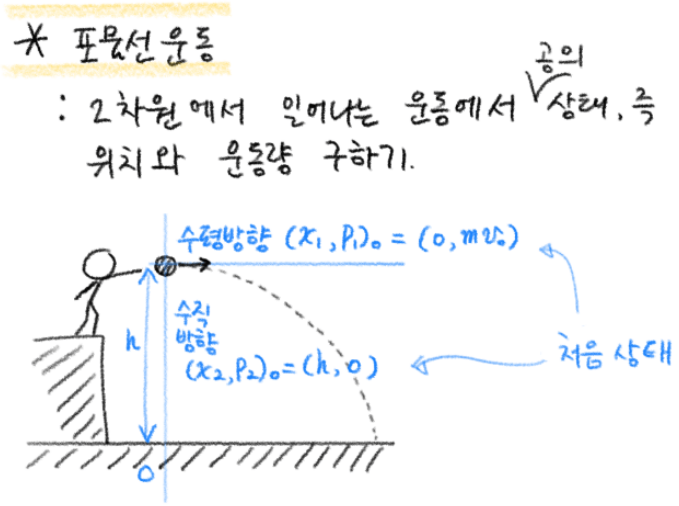
슈뢰딩거의 고양이를 삼킨 큐비트뱀 ^^;

01

양자 컴퓨팅을 위한

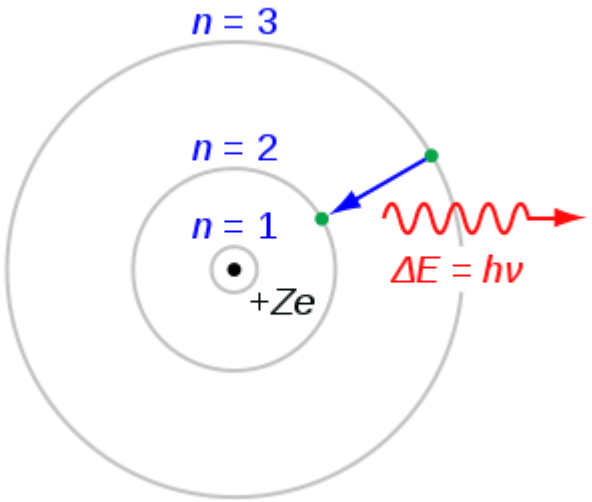
양자 역학

고전 역학 .vs. 양자 역학



$$F = ma$$

Classical Mechanics



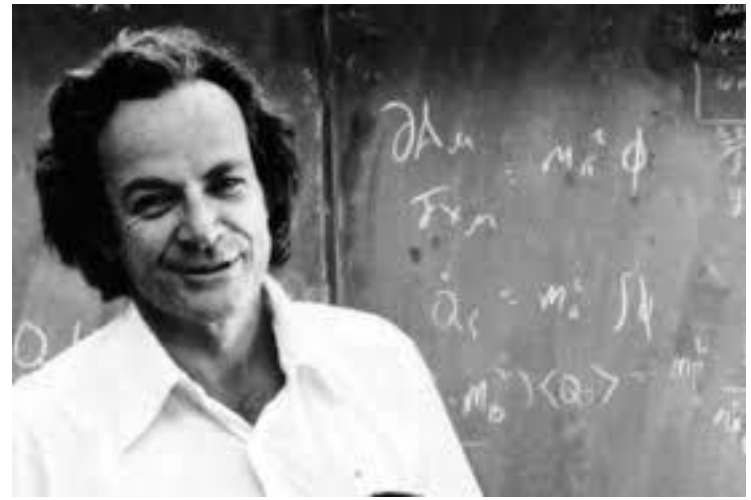
$$H\psi = E\psi$$

Quantum Mechanics

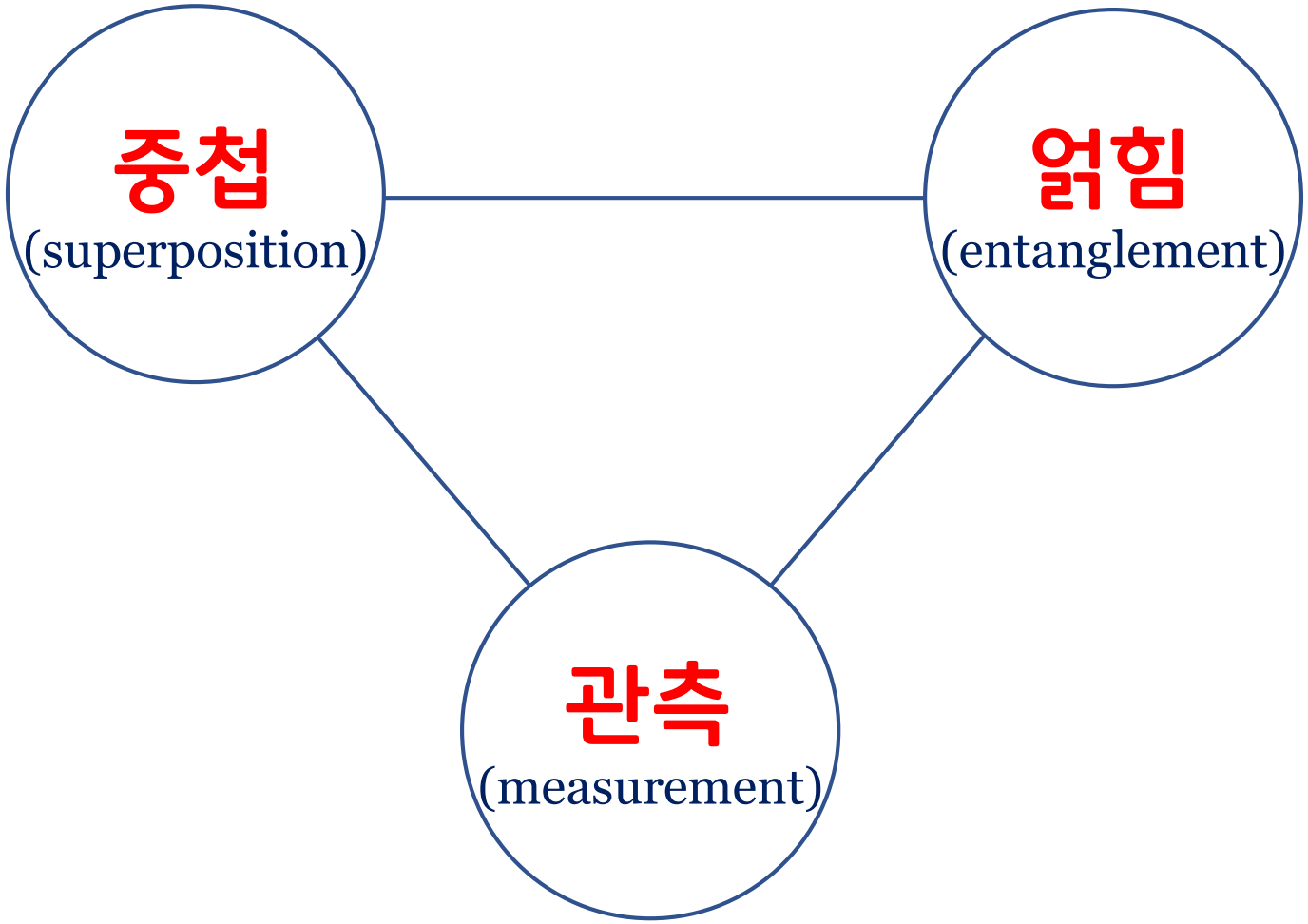
Q. 양자 컴퓨팅을 하려면 양자 역학을 알아야 하나요?

만약, 양자 역학을 접하고서도 심한 충격을 받지 않았다면
양자 역학을 제대로 이해하지 못했기 때문이다.
- 닐스 보어.

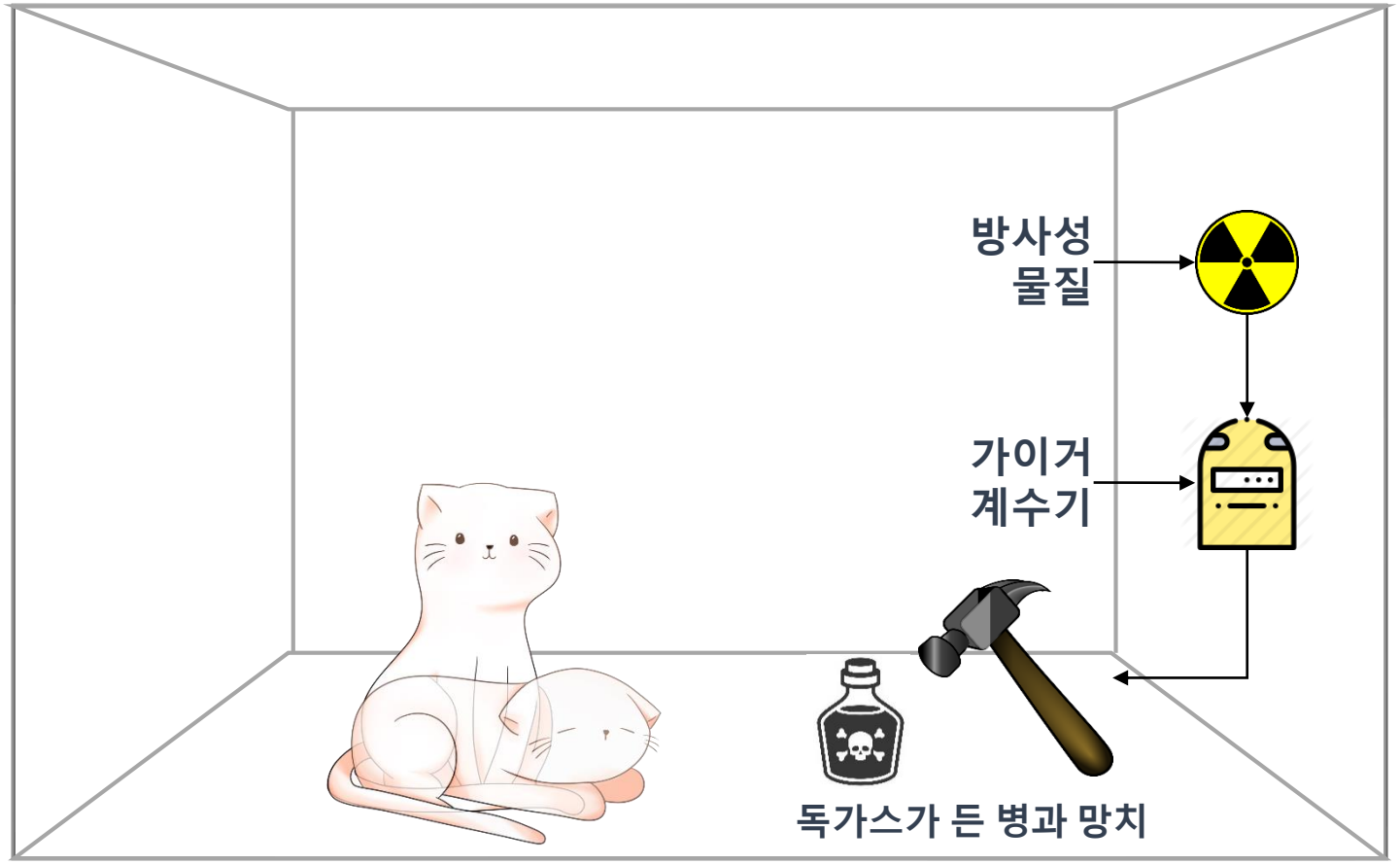
내 생각에, 양자 역학을 이해하는 사람은
아무도 없다고 자신 있게 말할 수 있다.
- 리처드 파인만.



A. 딱, 세 가지 양자 현상만 이해하고 넘어가자!

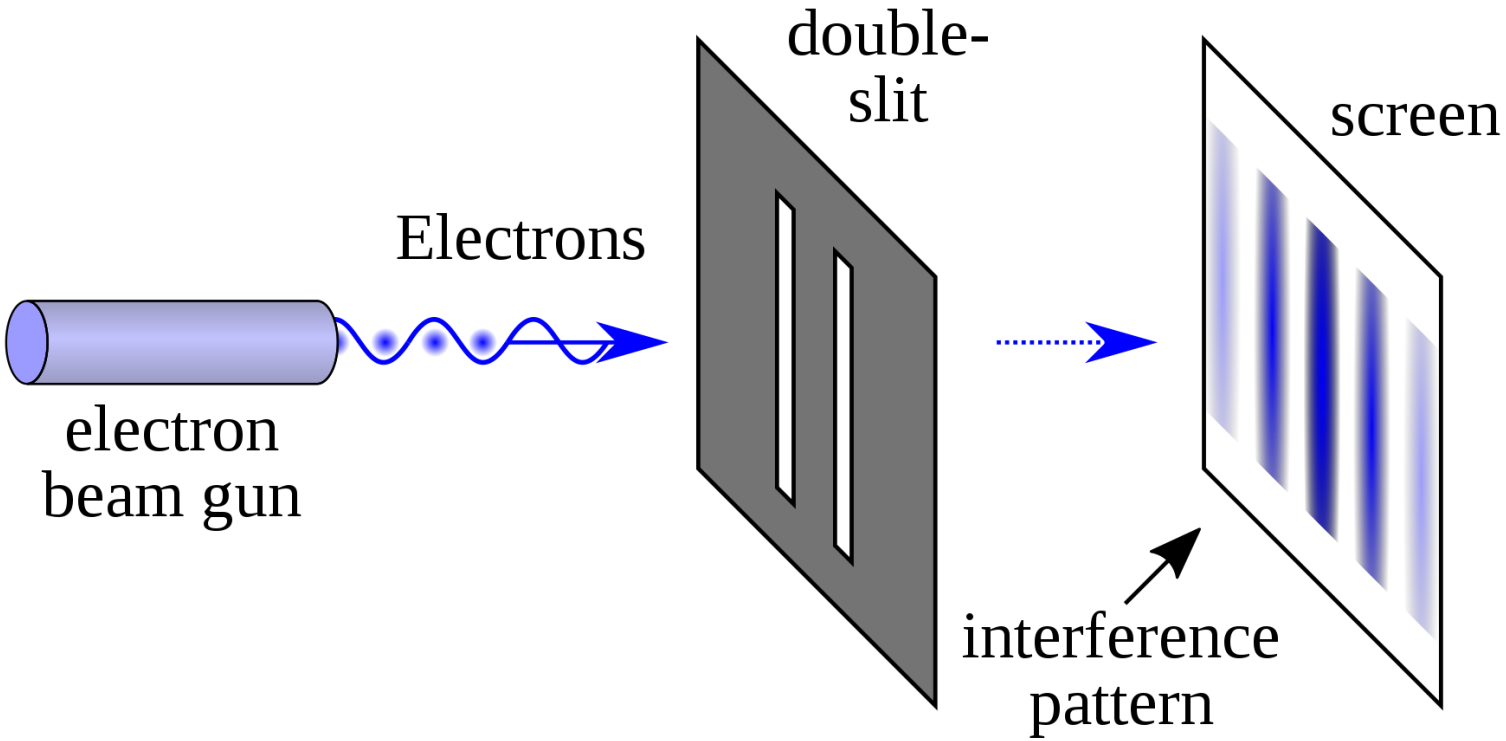


중첩: superposition



슈뢰딩거의 고양이

관측: measurement



이중 슬릿 실험

얽힘: entanglement

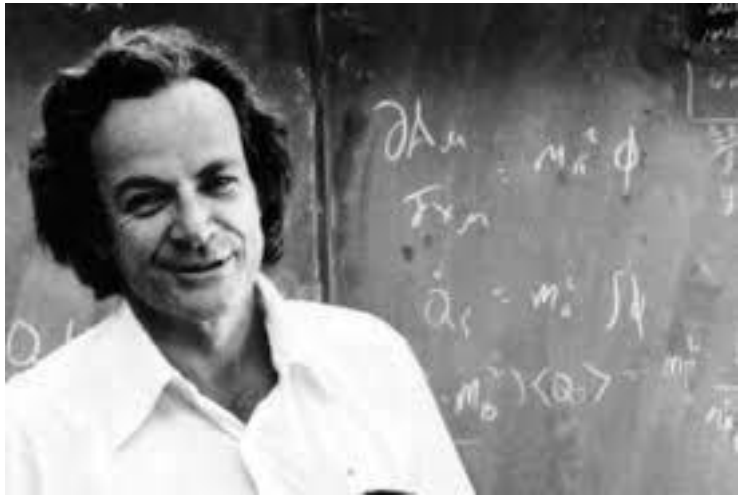
유령 같은 원격 작용

02

양자 컴퓨터가

뭐길래?

양자 컴퓨터의 탄생



Feynman, Richard P.

"*Simulating physics with computers.*"

Int. J. Theor. Phys 21.6/7 (1982).

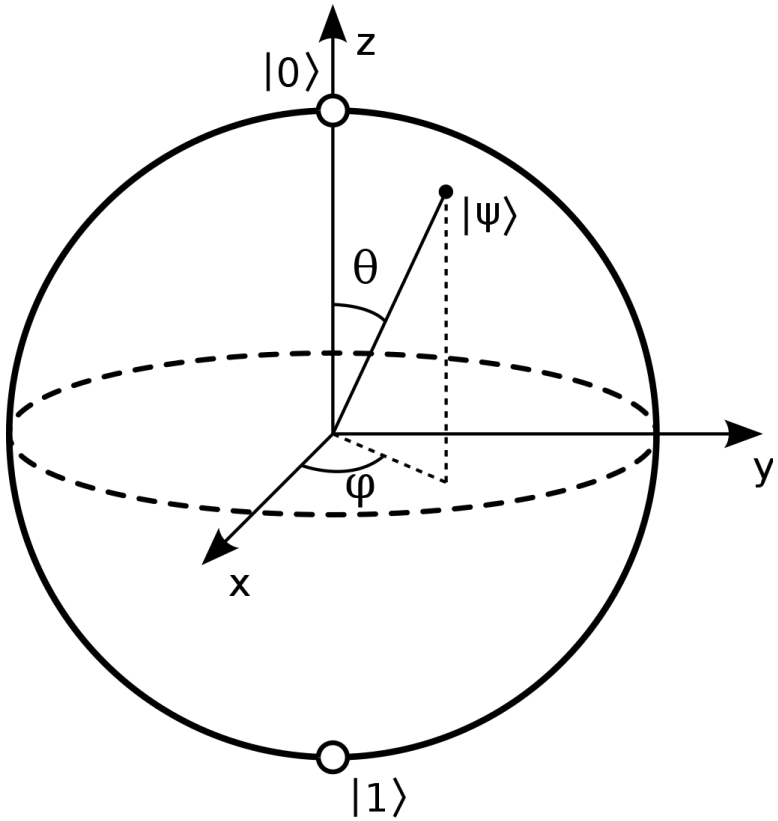
비트에서 큐비트로

bit: *binary digit*

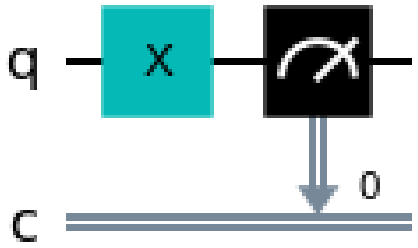
qubit: *quantum bit*

양자 상태와 큐비트

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$



양자 연산: 양자 상태의 변환

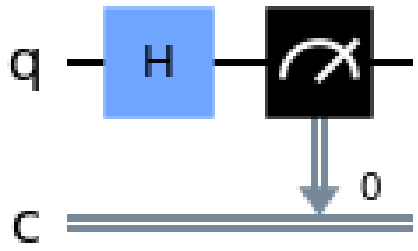


$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$X|\psi\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}$$

$$X|0\rangle = |1\rangle, X|1\rangle = |0\rangle$$

단일 큐비트의 중첩 상태: 하다마르(H) 게이트

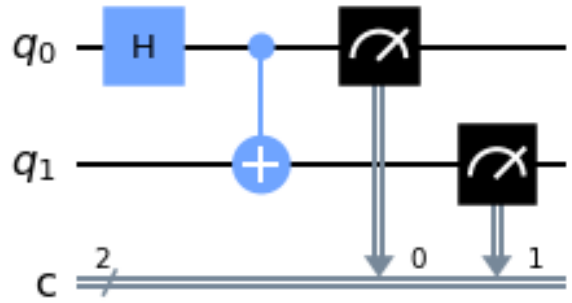


$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

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

다중 큐비트의 얽힘 상태: Controlled-NOT (CX) 게이트



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

$$CX|q_0 q_1\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

q_0	q_1
0	0
0	1
1	0
1	1

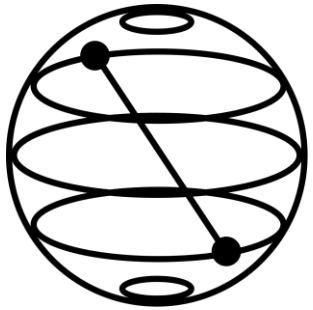
q_0	q_1
0	0
0	1
1	1
1	0

03

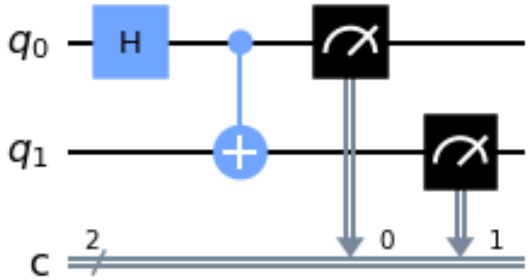
양자 알고리즘으로

할 수 있는 것들

양자 컴퓨터 프로그래밍: on *Qiskit* in *Jupyter* with *Python*



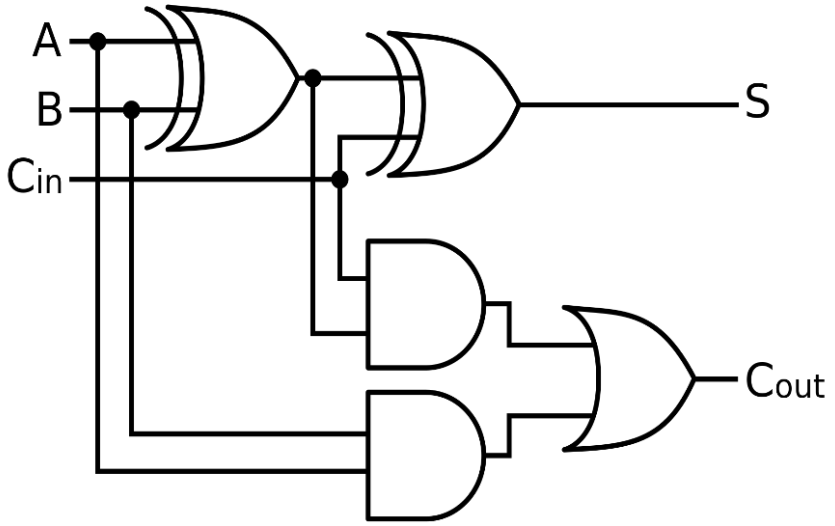
```
from qiskit import QuantumCircuit, execute, Aer
from qiskit.visualization import plot_histogram
circuit = QuantumCircuit(2, 2)
circuit.h(0)
circuit.cx(0, 1)
circuit.measure([0, 1], [0, 1])
circuit.draw()
```



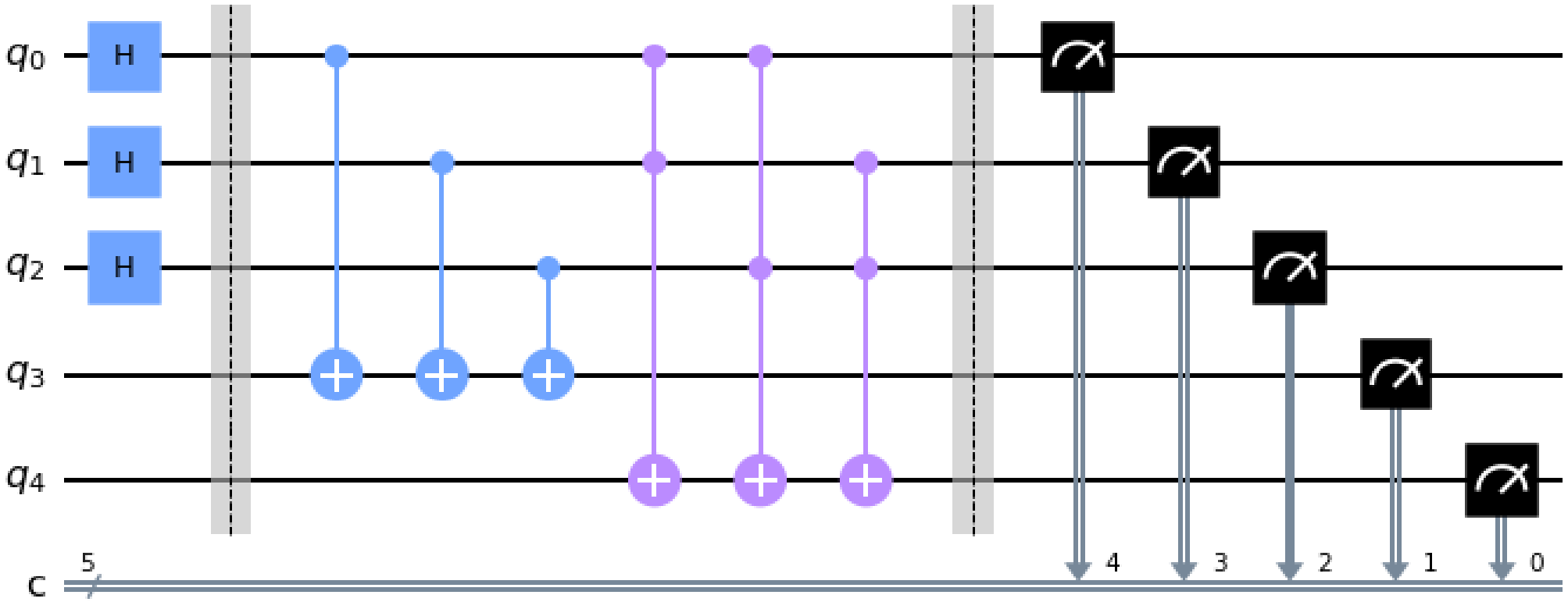
양자 컴퓨터로 덧셈 해보기

			1	1	1	0	0	0	0
				1	0	1	0	1	0
+	1	0	1	1	1	1	0	0
	1	1	1	0	0	1	1	0

carry



고전 컴퓨터에서의 덧셈과 전가산기

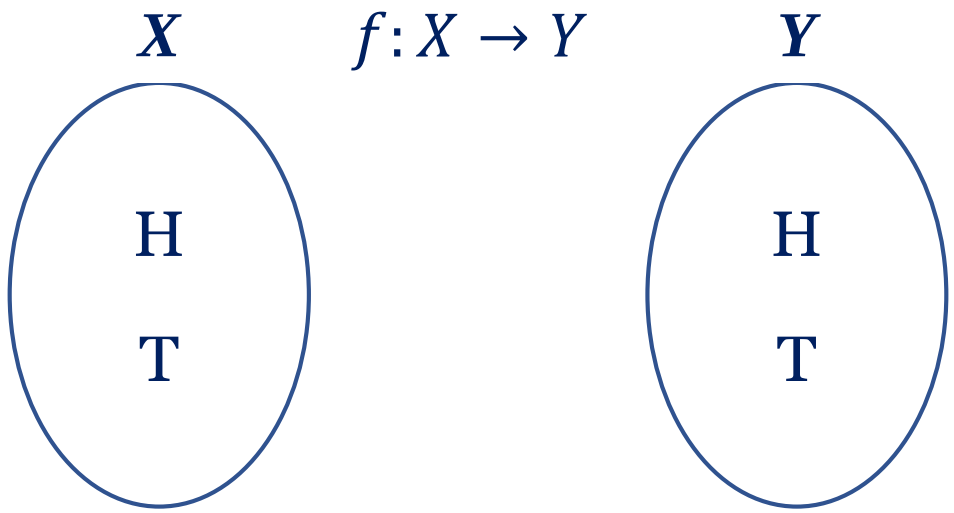


양자 컴퓨터에서의 덧셈을 위한 전가산기

양자 우월성: 마법의 동전 문제

- 어떤 동전에 마법이 걸려 있다.
- 이 동전은 앞면을 위에 놓고 던지느냐, 뒷면을 위에 놓고 던지느냐에 따라
  - 반드시 정해진 결과가 나오는 동전이다. (앞면이 나오거나, 뒷면이 나오거나)
- 이 동전을 던졌을 때, 그 결과가 항상 같은 지 알려면 (앞앞 or 뒤뒤)
  - 최소 몇 번을 던져봐야 하는가?

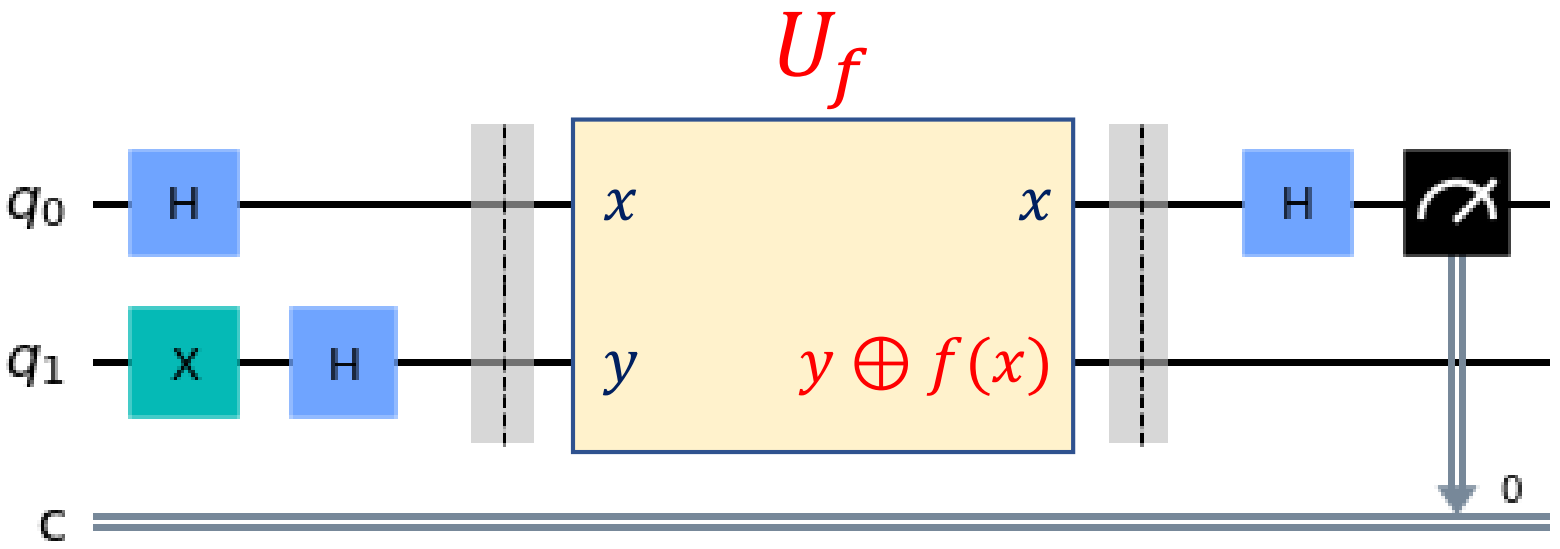
- 어떤 함수 f를 $f: X -> Y$ 라고 할 때,
  - 두 개의 서로 다른 입력 x, y가 주어질 때 $f(x)$와 $f(y)$는 같은가?
- 이 때, $X = \{H, T\}, Y = \{H, T\}$ 이다.



$$f(H) = f(T)?$$

$$f(H) \neq f(T)?$$

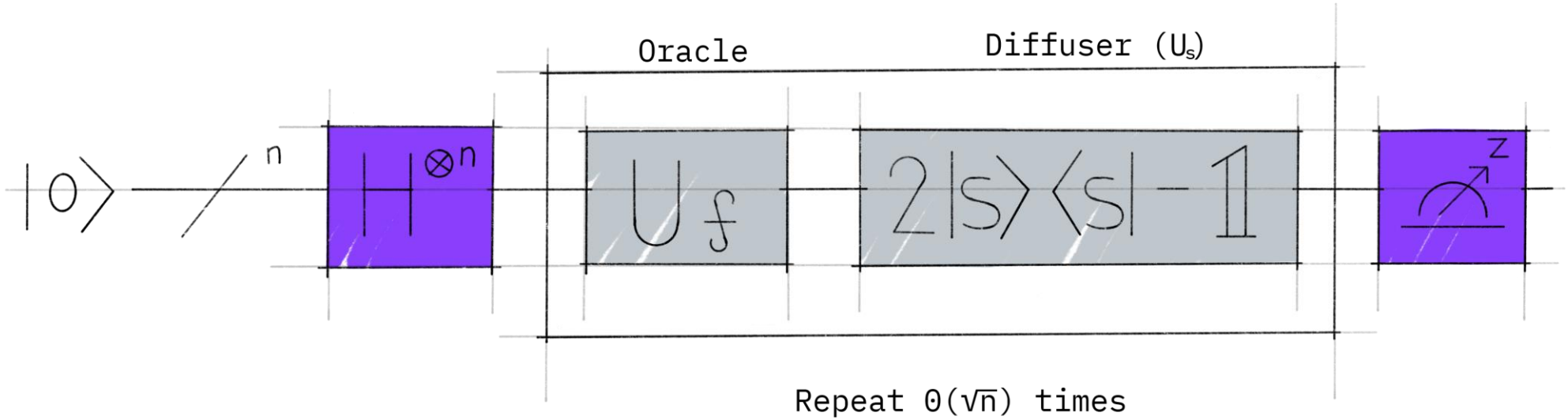
도이치 알고리즘: 한 번의 동전 던지기로 가능

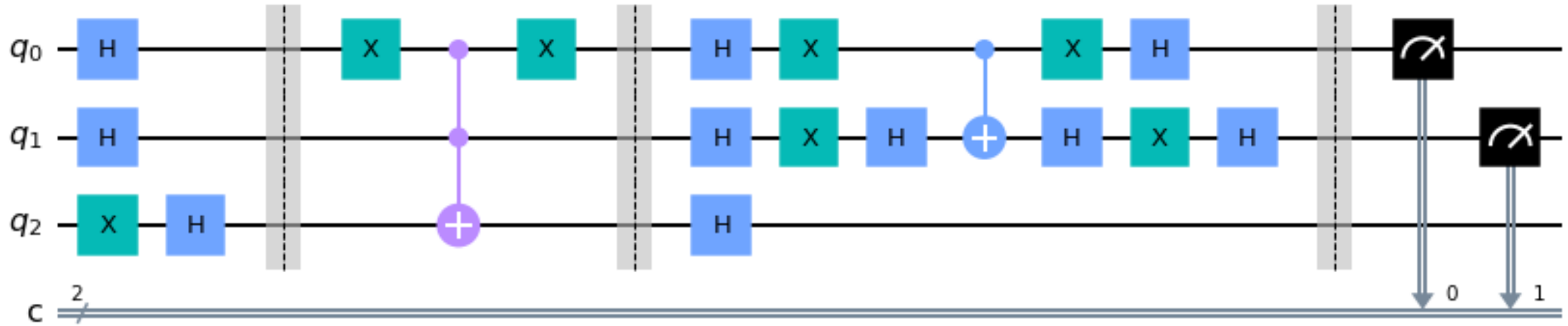


건초더미에서 바늘 찾기 문제

- 문제: n개의 데이터가 들어 있는 데이터베이스에서 x를 찾아라.
  - 비구조적 탐색 문제: *unstructured search* problem
  - 고전 알고리즘의 시간 복잡도: $O(n)$
- $O(n)$보다 더 빠른 양자 알고리즘이 존재할 수 있는가?
  - Lov Grover (1996): $O(\sqrt{n})$ 에 가능함.
  - 어? 그러면, NP-완전 문제들이 다항시간에 풀리겠는데... 우왕, 굳!

그로버 알고리즘: Finding a *needle* in a *haystack*.





소인수 분해 문제

- $N = p \times q$ 일 때, 매우 큰 소수 p, q에 대해서 소인수 분해 가능?
- 고전 컴퓨터로는 지수시간 복잡도를 극복할 수 없음
- RSA 암호화: 두 소수의 합성수 N을 이용해서 p, q를 공개키, 비밀키로 사용
- 다항시간에 소인수 분해 가능하면?: RSA 붕괴 $\rightarrow$ 지구 멸망

쇼어의 양자 알고리즘: 다항 시간에 소인수 분해 가능

- SHOR'S-FACTORING-ALGORITHM
  1. 1보다 크고 N보다 작은 정수 a를 임의로 선택
  2. 만일, $\gcd(N, a) \neq 1$이면, p를 찾았다!
     - 따라서, $p = \gcd(N, a), q = N/\gcd(N, a)$
  3. 함수 $f(x) = a^x \pmod N$의 주기(period) r을 찾는다.
     - 여기서 찾은 주기 r이 짝수가 아니면, 1번 단계부터 다시 시작한다.
  4. 주기 r로부터 두 개의 최대공약수 gcd_1, gcd_2 를 찾는다.
     - $gcd_1 = \gcd(N, a^{r/2} + 1), gcd_2 = \gcd(N, a^{r/2} - 1)$
  5. gcd_1, gcd_2이 1과 N이라면, 1번 단계부터 다시 시작한다.
     - 아니면, 마침내 소인수들을 찾았으므로, $p = gcd_1, q = gcd_2$ 리턴

$$N = 15 = 3 \times 5$$

- **1보다 크고 N보다 작은 정수 a를 임의로 선택**

$a = \{3, 5, 6, 9, 10, 12\}$ $a = \{2, 4, 7, 8, 11, 13, 14\}$

$\gcd(N, a) = \{3, 5, 3, 3, 5, 3\}$ $\gcd(N, a) = \{1, 1, 1, 1, 1, 1\}$

$$N = 15 = 3 \times 5$$

$a = \{2, 4, 7, 8, 11, 13, 14\}$

- **함수 $f(x) = a^x \pmod N$의 주기(period) r을 찾는다.**
- **여기서 찾은 주기 r이 짝수가 아니면, 1번 단계부터 다시 시작한다.**

$a = 2$: $f(0), f(1), f(2), f(3), f(4), f(5), \ldots$

$1 \pmod{15}, 2 \pmod{15}, 4 \pmod{15}, 8 \pmod{15}, 16 \pmod{15}, 32 \pmod{15}, \ldots$

$1, 2, 4, 8, 1, 2, 4, 8, 1, \cdots$

$$N = 15 = 3 \times 5$$

$a = 7$:

$$1 (\mathrm{mod}\, 15), 7 (\mathrm{mod}\, 15), 49 (\mathrm{mod}\, 15), 343 (\mathrm{mod}\, 15), 2401 (\mathrm{mod}\, 15), \cdots$$

$$1, 7, 4, 13, 1, 7, 4, 13, 1, 7, \cdots$$

$a = 4$:

$$1 (\mathrm{mod}\, 15), 4 (\mathrm{mod}\, 15), 16 (\mathrm{mod}\, 15), 64 (\mathrm{mod}\, 15), 256 (\mathrm{mod}\, 15), \cdots$$

$$1, 4, 1, 4, 1, 4, 1, 4, 1, 4, \cdots$$

$$N = 15 = 3 \times 5$$

- **주기 r로부터 두 개의 최대공약수 gcd_1, gcd_2 를 찾는다.**

$$gcd_1 = \gcd(N, a^{r/2} + 1),\ gcd_2 = \gcd(N, a^{r/2} - 1)$$

$a = 7,\ r = 4$:

$$gcd_1 = \gcd(15, 50) = 5$$

$$gcd_2 = \gcd(15, 48) = 3$$

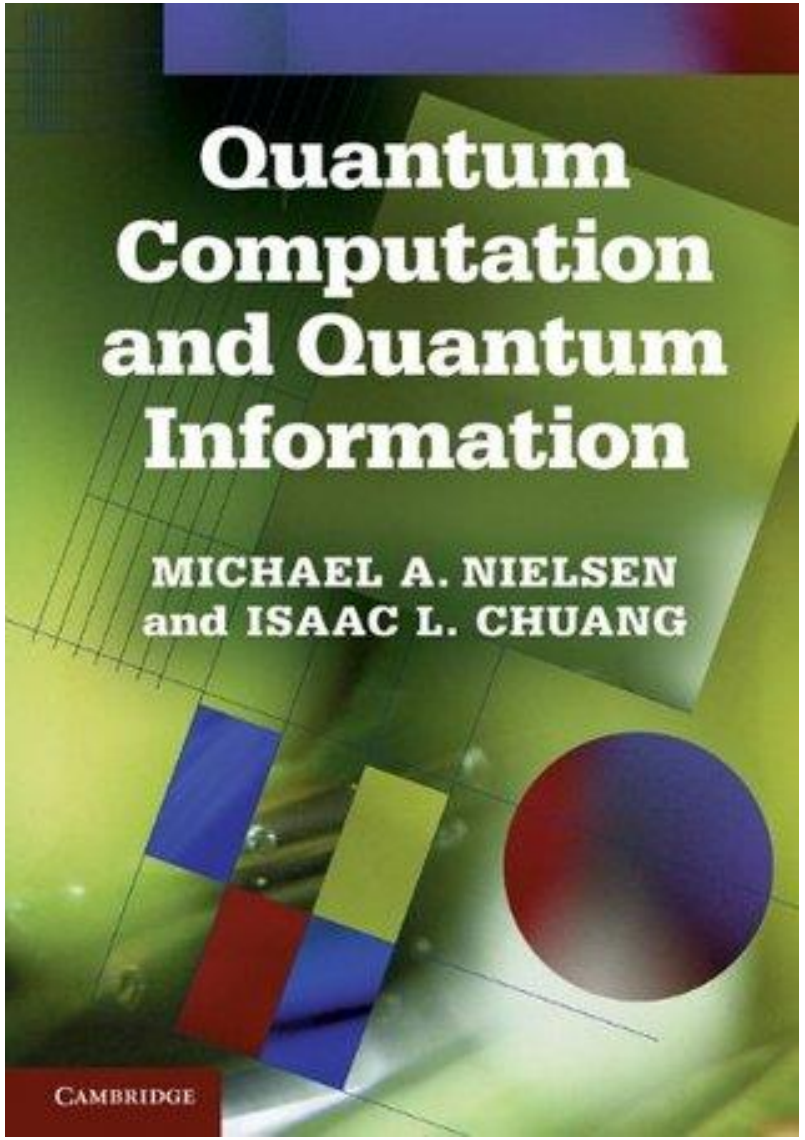
양자 알고리즘으로 할 수 있는 더 재밌는 것들…

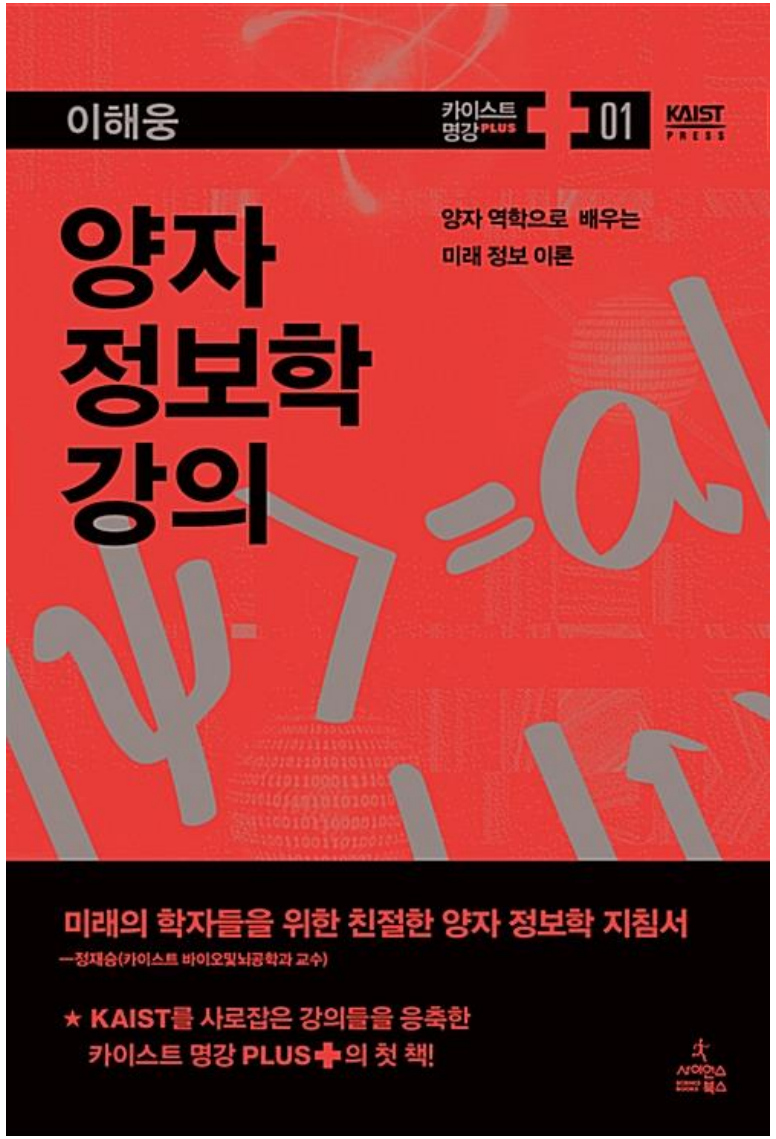
- Breaking the Bitcoin?
- Quantum Cryptography
- Quantum Telecommunication
- Quantum Teleportation
- Quantum Machine Learning
- And So Force…

주니온TV@youtube

어서와! 양자컴퓨팅은 처음이지?

진지한 양자 컴퓨팅 공부를 위한 책 소개





Any Questions?

주니온TV@Youtube

자세히 보면 유익한 코딩 채널