Introduction to Quantum Computing

How I Learned to Stop Worrying and Love the Bomb
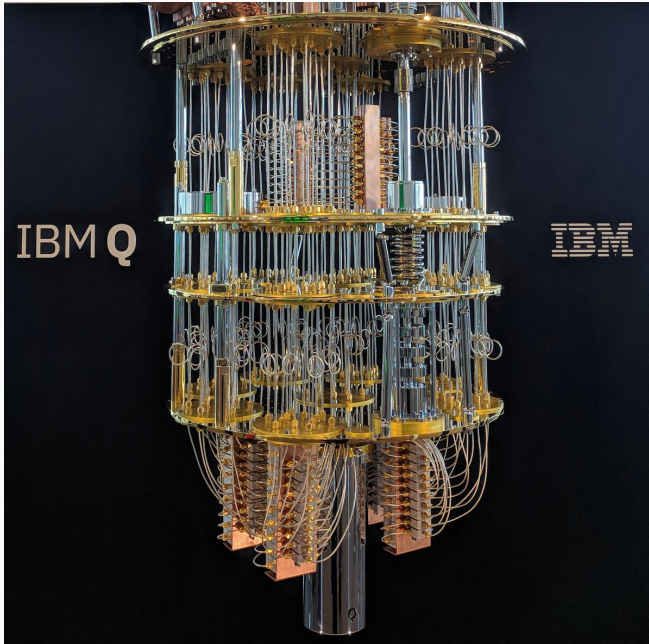
Dr. Omri Har-Shemesh

13. February 2019

Schmiede.ONE GmbH & Co. KG

Introduction

- New computing paradigm

Big big picture

- New computing paradigm

- Uses the rules of quantum mechanics

- New computing paradigm

- Uses the rules of quantum mechanics

- Might be able to solve some problems exponentially faster

- New computing paradigm

- Uses the rules of quantum mechanics

- Might be able to solve some problems exponentially faster

- Definitely could simulate quantum systems better

- New computing paradigm

- Uses the rules of quantum mechanics

- Might be able to solve some problems exponentially faster

- Definitely could simulate quantum systems better

- Realizable in large scale?

- New computing paradigm

- Uses the rules of quantum mechanics

- Might be able to solve some problems exponentially faster

- Definitely could simulate quantum systems better

- Realizable in large scale?

- ¯_(ツ)_/¯

- Quantum Mechanics is technical

- Quantum Mechanics is technical

- Quantum Mechanics is weird

- Quantum Mechanics is technical

- Quantum Mechanics is weird

- Quantum Computing is technical

- Quantum Mechanics is technical

- Quantum Mechanics is weird

- Quantum Computing is technical

- Quantum Algorithms are hard

Examples?

Breaking the RSA algorithm
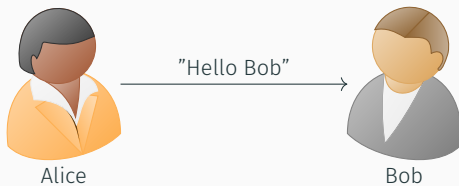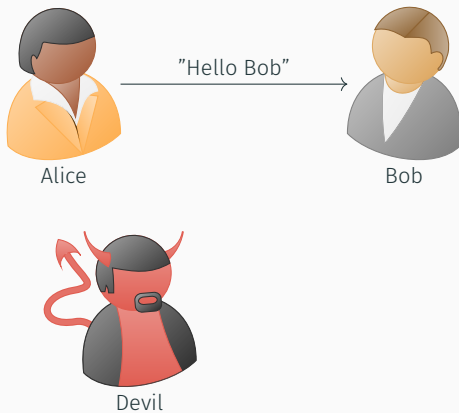
Alice

Alice

Bob

Alice

Bob

Generate Public/Private keys

$$e, d, N = pq$$

"Public Key: e, N"

Alice

Bob
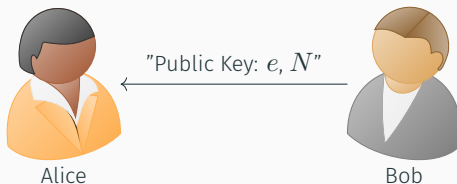
Generate Public/Private keys

$$e, d, N = pq$$

Alice

Bob

$$m = \text{encode}(\text{"Hello World"})$$
$$c = m^e \pmod{N}$$

$$m = \text{encode}(\text{"Hello World"})$$
$$c = m^e \ (\text{mod } N)$$

$m = \text{encode}("\text{Hello World}")$
$c = m^e \,(\text{mod } N)$

$m = c^d \,(\text{mod } N)$
$\text{decode}(m)$

It's easy to find $e, d, N \in \mathbb{N}$, such that:

It's easy to find $e, d, N \in \mathbb{N}$, such that:

$$\boxed{(m^e)^d \equiv m \pmod{N}}$$

It's easy to find $e, d, N \in \mathbb{N}$, such that:

$$\boxed{(m^e)^d \equiv m \pmod{N}}$$

- $N = pq$ with p, q large prime numbers.

It's easy to find $e, d, N \in \mathbb{N}$, such that:

$$\boxed{(m^e)^d \equiv m \pmod{N}}$$

- $N = pq$ with p, q large prime numbers.

- e, N - Public Key

It's easy to find $e, d, N \in \mathbb{N}$, such that:

$$\boxed{(m^e)^d \equiv m \pmod{N}}$$

- $N = pq$ with p, q large prime numbers.

- e, N - Public Key

- d, N - Private Key

It's easy to find $e, d, N \in \mathbb{N}$, such that:

$$\boxed{(m^e)^d \equiv m \,(\mathrm{mod}\, N)}$$

- $N = pq$ with p, q large prime numbers.

- e, N - Public Key

- d, N - Private Key

It's easy to find $e, d, N \in \mathbb{N}$, such that:

$$\boxed{(m^e)^d \equiv m \ (\mathrm{mod} \ N)}$$

- $N = pq$ with p, q large prime numbers.

- e, N - Public Key

- d, N - Private Key

But it's hard to find p, q, such that $pq = N$.

Factorizing on a Classical Computer

Bits	Time	Notes
128	less than 2 seconds	
192	16 seconds	
256	35 minutes	
260	1 hour	
512	73 days	in 2009

Factorizing on a Classical Computer

Bits	Time	Notes
128	less than 2 seconds	
192	16 seconds	
256	35 minutes	
260	1 hour	
512	73 days	in 2009
768	1500 CPU years	Largest known (2010), took 2 years on hundreds of computers

Factorizing on a Classical Computer

Bits	Time	Notes
128	less than 2 seconds	
192	16 seconds	
256	35 minutes	
260	1 hour	
512	73 days	in 2009
768	1500 CPU years	Largest known (2010), took 2 years on hundreds of computers

Most RSA implementations use between 1024 and 4096 bits.

Would require ~ 4000 qubits to break 2048-bit RSA

Movie

The Technical Part

"Shut up and calculate"
_____
David Mermin

- Representing computation with linear algebra

- Qubits, superposition and quantum logic gates

- Simplest problem where a quantum computer outperforms a classical one

- Bonus: Quantum entanglement and quantum teleportation

One bit with value 0, also written as $|0\rangle$ (Dirac vector notation)

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

One bit with value 1, also written as $|1\rangle$

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} ax + by \\ cx + dy \end{pmatrix}$$

$$\begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} ax + by + cz \\ dx + ey + fz \\ gx + hy + iz \end{pmatrix}$$

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} w & x \\ y & z \end{pmatrix} = \begin{pmatrix} aw + by & ax + bz \\ cw + dy & cx + dz \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}$$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

What are the operations we can perform on one classical bit?

Operations on one classical bit (cbit)

Identity $\qquad f(x) = x \qquad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

Negation $\qquad f(x) = \neg x \qquad \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Constant-0 $\qquad f(x) = 0 \qquad \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Constant-1 $\qquad f(x) = 1 \qquad \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

Reversible computing

- Given the operation and the input, you can always infer the output.
  - For $Ax = b$, given b and A, you can uniquely find x.

- Given the operation and the input, you can always infer the output.
  - For $Ax = b$, given b and A, you can uniquely find x.

- Permutations are reversible; erasing and overwriting are not
  - Identity and negation are reversible.
  - Constant-0 and Constant-1 are not reversible.

Reversible computing

- Given the operation and the input, you can always infer the output.
  - For $Ax = b$, given b and A, you can uniquely find x.

- Permutations are reversible; erasing and overwriting are not
  - Identity and negation are reversible.
  - Constant-0 and Constant-1 are not reversible.

- Quantum computers use only reversible operations.
  - In fact, all quantum operations are their own inverse.

$$\begin{pmatrix} x_0 \\ x_1 \end{pmatrix} \otimes \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} = \begin{pmatrix} x_0 \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} \\ x_1 \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} x_0 y_0 \\ x_0 y_1 \\ x_1 y_0 \\ x_1 y_1 \end{pmatrix} \qquad \begin{pmatrix} 1 \\ 2 \end{pmatrix} \otimes \begin{pmatrix} 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 3 \\ 4 \\ 6 \\ 8 \end{pmatrix}$$

$$\begin{pmatrix} x_0 \\ x_1 \end{pmatrix} \otimes \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} \otimes \begin{pmatrix} z_0 \\ z_1 \end{pmatrix} = \begin{pmatrix} x_0 y_0 z_0 \\ x_0 y_0 z_1 \\ x_0 y_1 z_0 \\ x_0 y_1 z_1 \\ x_1 y_0 z_0 \\ x_1 y_0 z_1 \\ x_1 y_1 z_0 \\ x_1 y_1 z_1 \end{pmatrix} \qquad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \qquad |01\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$|100\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$|10\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \qquad |11\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

- The tensor representation is called the product state.

- It can be factored back into the individual state representation.

- The product state of n bits is a vector of size 2^n.

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- If most significant bit is control, and least-significant is target, then:

  $|00\rangle$ $|00\rangle$

  $|01\rangle$ $|01\rangle$

  $|10\rangle$ $|10\rangle$

  $|11\rangle$ $|11\rangle$

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- If most significant bit is control, and least-significant is target, then:

$$|00\rangle \longrightarrow |00\rangle$$
$$|01\rangle \qquad |01\rangle$$
$$|10\rangle \qquad |10\rangle$$
$$|11\rangle \qquad |11\rangle$$

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- If most significant bit is control, and least-significant is target, then:

$$|00\rangle \longrightarrow |00\rangle$$

$$|01\rangle \longrightarrow |01\rangle$$

$$|10\rangle \qquad |10\rangle$$

$$|11\rangle \qquad |11\rangle$$

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- If most significant bit is control, and least-significant is target, then:

$$|00\rangle \longrightarrow |00\rangle$$
$$|01\rangle \longrightarrow |01\rangle$$
$$|10\rangle \qquad |10\rangle$$
$$|11\rangle \qquad |11\rangle$$

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- If most significant bit is control, and least-significant is target, then:

$$|00\rangle \longrightarrow |00\rangle$$

$$|01\rangle \longrightarrow |01\rangle$$

$$|10\rangle |10\rangle$$

$$|11\rangle |11\rangle$$

- Takes two bits, one control bit and one target bit.

- If the control bit is set, flip the target bit, otherwise leave it.

- If most significant bit is control, and least-significant is target, then:

$$|00\rangle \longrightarrow |00\rangle$$
$$|01\rangle \longrightarrow |01\rangle$$
$$|10\rangle \quad\quad |10\rangle$$
$$|11\rangle \quad\quad |11\rangle$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$C \left| 10 \right\rangle = C \left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \left| 11 \right\rangle$$

$$C \left| 11 \right\rangle = C \left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \left| 10 \right\rangle$$

$$C\left|00\right\rangle = C\left(\begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix}\right) = \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}\begin{pmatrix}1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \left|00\right\rangle$$

$$C\left|01\right\rangle = C\left(\begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix}\right) = \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}\begin{pmatrix}0\\1\\0\\0\end{pmatrix} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}0\\1\end{pmatrix} = \left|01\right\rangle$$

- Cbits are a special case of Qubits!

- A qubit is represented by $\begin{pmatrix} a \\ b \end{pmatrix}$ where a and b are complex numbers such that $||a||^2 + ||b||^2 = 1$.

  - The cbit vectors $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ fit this definition.

- Example qubit values:

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \qquad \begin{pmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{pmatrix} \qquad \begin{pmatrix} -1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix}$$

What does that mean?

What does that mean?

Superposition

The qubit is in a state of both $|0\rangle$ and $|1\rangle$. We can write this as:

$$\begin{pmatrix} a \\ b \end{pmatrix} = a \begin{pmatrix} 1 \\ 0 \end{pmatrix} + b \begin{pmatrix} 0 \\ 1 \end{pmatrix} = a |0\rangle + b |1\rangle.$$
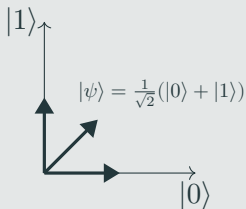
What does that mean?

Superposition

The qubit is in a state of both $|0\rangle$ and $|1\rangle$. We can write this as:

$$\begin{pmatrix} a \\ b \end{pmatrix} = a \begin{pmatrix} 1 \\ 0 \end{pmatrix} + b \begin{pmatrix} 0 \\ 1 \end{pmatrix} = a |0\rangle + b |1\rangle .$$

$|1\rangle$

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

$|0\rangle$

What does that mean?

Superposition

The qubit is in a state of both $|0\rangle$ and $|1\rangle$. We can write this as:

$$\begin{pmatrix} a \\ b \end{pmatrix} = a \begin{pmatrix} 1 \\ 0 \end{pmatrix} + b \begin{pmatrix} 0 \\ 1 \end{pmatrix} = a |0\rangle + b |1\rangle .$$

Amplitudes

a and b are called amplitudes. $||a||^2$ is the probability of the qubit being 0 when measured; $||b||^2$ is the probability of measuring 1.

What does that mean?

Superposition

The qubit is in a state of both $|0\rangle$ and $|1\rangle$. We can write this as:

$$\begin{pmatrix} a \\ b \end{pmatrix} = a \begin{pmatrix} 1 \\ 0 \end{pmatrix} + b \begin{pmatrix} 0 \\ 1 \end{pmatrix} = a |0\rangle + b |1\rangle .$$

Amplitudes

a and b are called amplitudes. $||a||^2$ is the probability of the qubit being 0 when measured; $||b||^2$ is the probability of measuring 1.

Measurement

The measurement of the qubit collapses its state. It will be in the state $|0\rangle$ if we measured 0 and $|1\rangle$ if we measured $1^{\dagger}$.

For example

The qubit $\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$ has a $\left\| \frac{1}{\sqrt{2}} \right\|^2 = \frac{1}{2}$ chance of collapsing to $|0\rangle$ or $|1\rangle$.

The qubit $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ has 100% chance of collapsing to $|0\rangle$, and $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ has a 100% chance of collapsing to $|1\rangle$.

- Multiple qubits are represented by the tensor product:

$$\begin{pmatrix} a \\ b \end{pmatrix} \otimes \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} ab \\ ad \\ bc \\ bd \end{pmatrix} \text{ with } ||ac||^2 + ||ad||^2 + ||bc||^2 + ||bd||^2 = 1.$$

- Multiple qubits are represented by the tensor product:

$$\begin{pmatrix} a \\ b \end{pmatrix} \otimes \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} ab \\ ad \\ bc \\ bd \end{pmatrix} \text{ with } ||ac||^2 + ||ad||^2 + ||bc||^2 + ||bd||^2 = 1.$$

- For example:

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}; \quad \left|\left|\frac{1}{2}\right|\right|^2 = \frac{1}{4}; \quad \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} = 1$$

- Multiple qubits are represented by the tensor product:

$$\begin{pmatrix} a \\ b \end{pmatrix} \otimes \begin{pmatrix} c \\ d \end{pmatrix} = \begin{pmatrix} ab \\ ad \\ bc \\ bd \end{pmatrix} \text{ with } ||ac||^2 + ||ad||^2 + ||bc||^2 + ||bd||^2 = 1.$$

- For example:

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} ; \quad \left\| \frac{1}{2} \right\|^2 = \frac{1}{4}; \quad \frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4} = 1$$

$\rightarrow$ There's an equal chance (25%) of measuring $|00\rangle, |01\rangle, |10\rangle, |11\rangle$.

- We operate on qubits in the same way as on cbits: with matrices.

- All the operations we saw so far (bit flip, CNOT, etc...) work on qubits as well.

- In reality, the matrix operations model some device that manipulates the real qubits without measurement.

- Some gates only make sense in the quantum context...

- The Hadaramd gate puts a $|0\rangle$ or $|1\rangle$ bit into exact superposition:
  $H |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$ and $H |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$.

- The Hadaramd gate puts a $|0\rangle$ or $|1\rangle$ bit into exact superposition:
  $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

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

- The Hadaramd gate puts a $|0\rangle$ or $|1\rangle$ bit into exact superposition:
  $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

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

- Note that $H^2 = HH = \mathbf{1}$ so $H^2|0\rangle = |0\rangle$ and $H^2|1\rangle = |1\rangle$.

- The Hadaramd gate puts a $|0\rangle$ or $|1\rangle$ bit into exact superposition: $H |0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$ and $H |1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$.

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

- Note that $H^2 = HH = \mathbf{1}$ so $H^2 |0\rangle = |0\rangle$ and $H^2 |1\rangle = |1\rangle$.

- This allows us to get out of superposition without measurement! So we can structure computations deterministically.

The Deutsch Oracle

Input ——————— Black Box ———————→ Output

- The Black Box is a deterministic function of one bit.

Input ———— [Black Box] ————→ Output

- The Black Box is a deterministic function of one bit.

- You can give it whatever input you want, and observe the output.

Input ——————— | Black Box | ———————→ Output

- The Black Box is a deterministic function of one bit.

- You can give it whatever input you want, and observe the output.

- How many queries do you need to determine the function on classical computer?

Input ──────── Black Box ────────→ Output

- The Black Box is a deterministic function of one bit.

- You can give it whatever input you want, and observe the output.

- How many queries do you need to determine the function on classical computer?

- And on a quantum computer?

Input ——————— Black Box ————→ Output

- What if we want to determine whether the function is constant or variable?

Input ———————— | Black Box | ————————→ Output

- What if we want to determine whether the function is constant or variable?

- How many queries on a classical computer?

Input ——————— Black Box ——————→ Output

- What if we want to determine whether the function is constant or variable?

- How many queries on a classical computer?

- And on a quantum computer?

Problem: the `constant-0` and `constant-1` functions are non-reversible.

Before we can continue, we have to write them in a reversible way:

Problem: the `constant-0` and `constant-1` functions are non-reversible.

Before we can continue, we have to write them in a reversible way:

Before:

Input $\xrightarrow{\;|x\rangle\;}$ [BB] $\xrightarrow{\;f(|x\rangle)\;}$ Output

Problem: the `constant-0` and `constant-1` functions are non-reversible.

Before we can continue, we have to write them in a reversible way:

Before:

Input ——$|x\rangle$—— **BB** ——$f(|x\rangle)$—→ Output

After:

Output ——$|0\rangle$—— **BB** ——$f(|x\rangle)$—→ Output'

Input ——$|x\rangle$—— ——$|x\rangle$—→ Input'
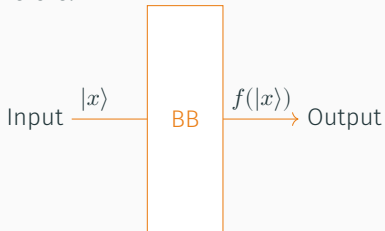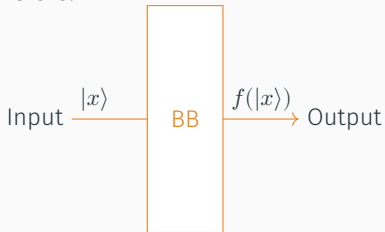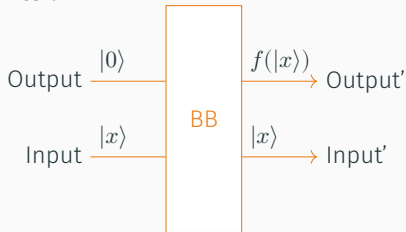
Problem: the `constant-0` and `constant-1` functions are non-reversible.

Before we can continue, we have to write them in a reversible way:

Before:
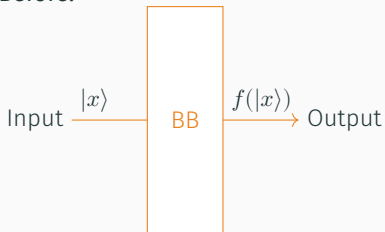
After:
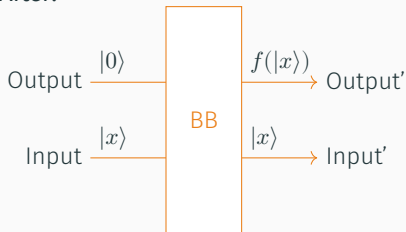
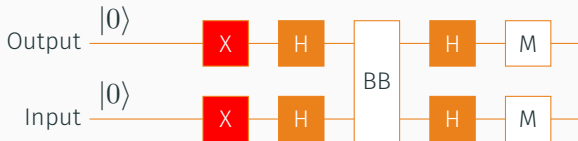The black box leaves the input qubit unchanged, writing the output of the function to the output qubit.
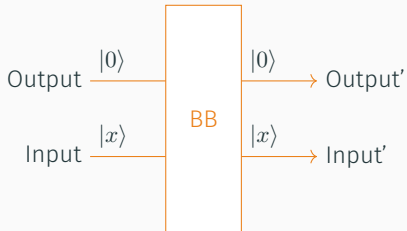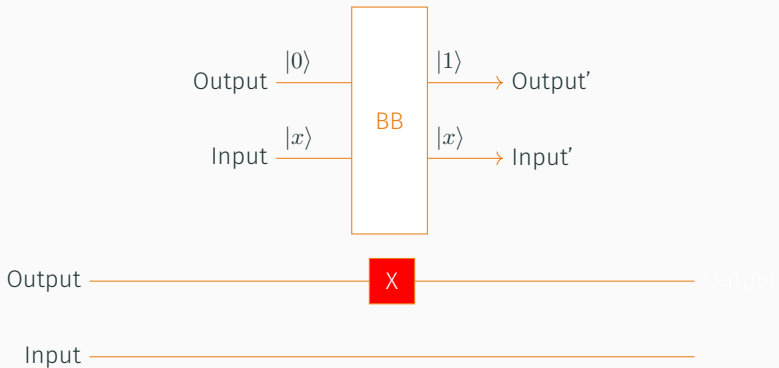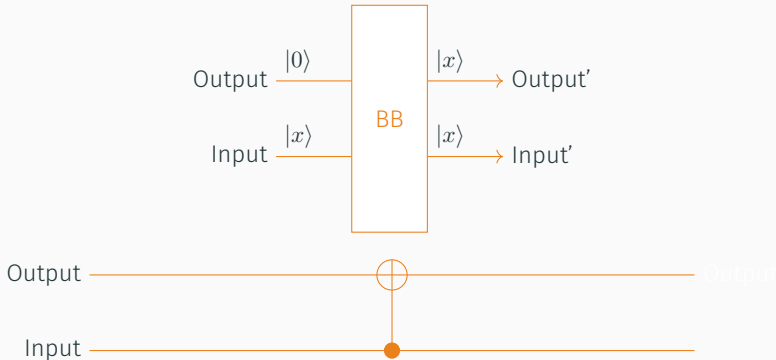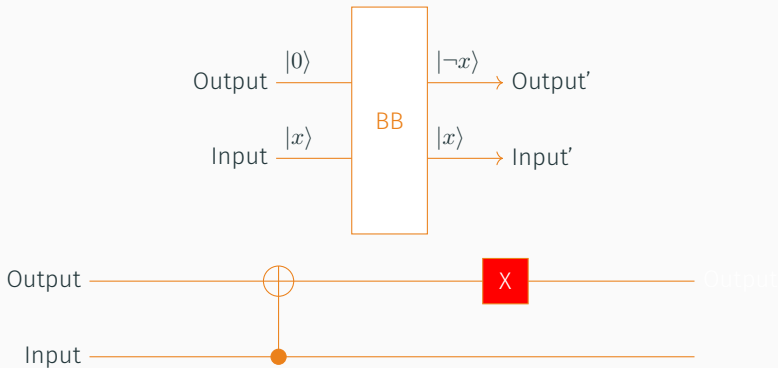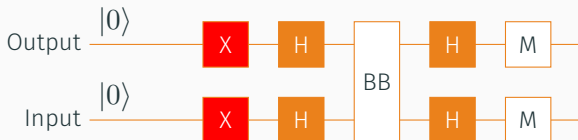
- So how do we solve the problem in one operation?
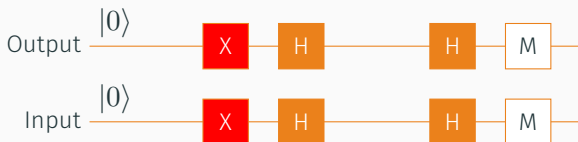
- So how do we solve the problem in one operation?

- So how do we solve the problem in one operation?

- If the black-box function is constant, we will measure $|11\rangle$.
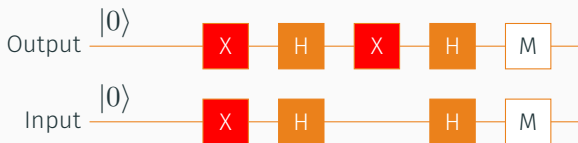- If the black-box function is variable, we will measure $|01\rangle$.

$$\text{Output'} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\text{Input'} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$
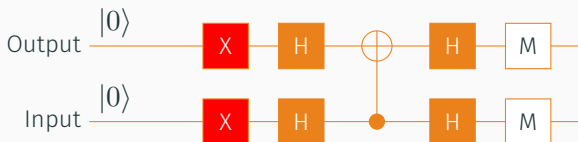
Output is $|11\rangle$.

$$\text{Output'} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\text{Input'} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$
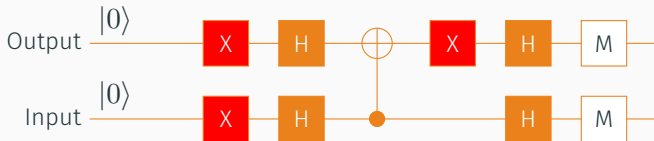
Output is $|11\rangle$.

$$C\left(\begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix}\right) = C\begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ -1 \\ -1 \\ 1 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \end{pmatrix}$$

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

Output (after applying Hadamard gate): $|01\rangle$.

Left as an exercise :)

Output: $|01\rangle$.

- We managed to determine whether the function is constant in one query!

- We magnified the difference between categories (CNOT gate) and neutralized the difference within the categories (NOT gate).

- There is a generalization of this to n-bit black boxes (Deutsch-Josza problem).

- It was an inspiration for Shor's algorithm!

Bonus Topic - Quantum Entanglement

- If the product state of two qubits cannot be factored, they are said to be entangled

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix} \otimes \begin{pmatrix} c \\ d \end{pmatrix}$$

$$ac = \frac{1}{\sqrt{2}}$$

$$ad = 0$$

$$bc = 0$$

$$bd = \frac{1}{\sqrt{2}}$$

- The system of equations has no solution, so we cannot factor the quantum state.

- This state has a 50% chance of collapsing to $|00\rangle$ and 50% chance of collapsing to $|11\rangle$.

To reach an entangled state is quite simple:

$$CH_1 \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right) = C \left(\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

- The qubits are forced to be equal (both are either 0 or 1)!

- If we measure one of them, the state of the other will collapse to be equal to the one we measured.

- This can happen even if they are very very far away from each other!

- The value is not predetermined.

- We cannot use this to transmit information, however.