Canary Wharf .NET User Group

CWNUG

Johnny Hooyberghs

Microsoft Q# and Azure Quantum

involved

involved

Johnny Hooyberghs

@djohnnieke

github.com/Djohnnie

www.involved-it.be

johnny.hooyberghs@involved-it.be

www.cvoantwerpen.be

slido

Join at

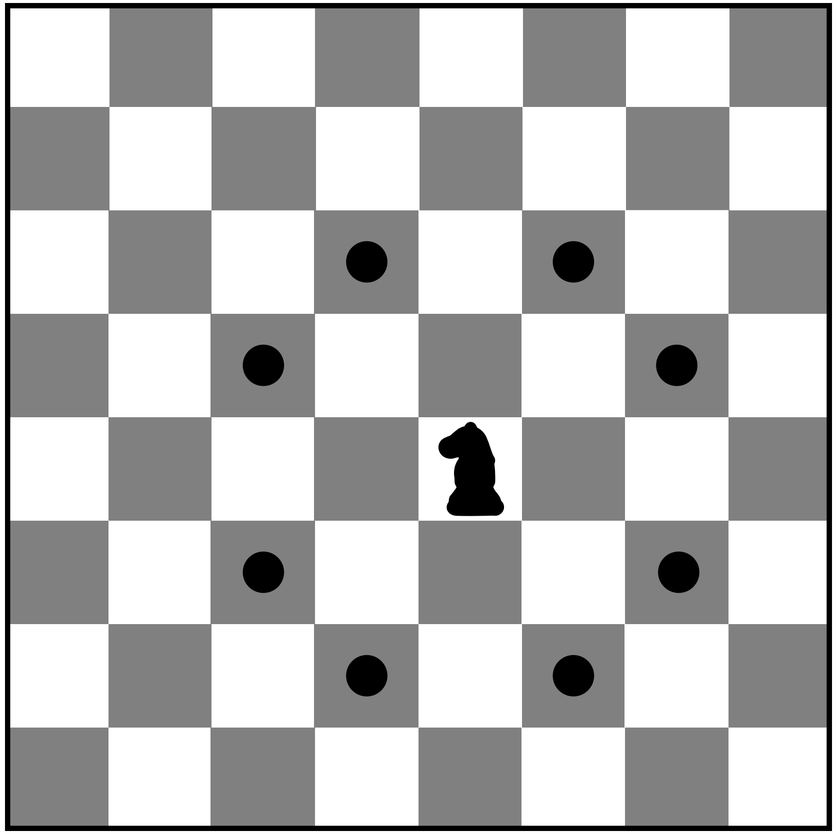
slido.com

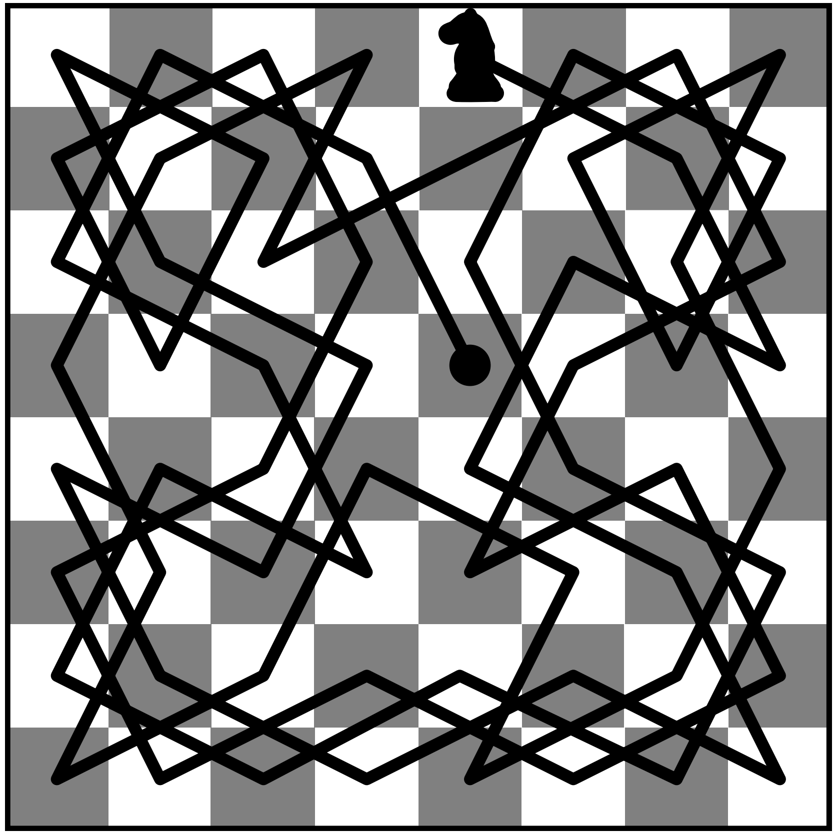
#CWDNUG

Why Quantum Computing?

- There are still a lot of problems that cannot be solved by computers
- CPU’s have their physical limits
- Current classical computing architectures already have issues with unwanted quantum side effects because of their scale
- Why try to simulate a complex quantum world using classical computers?

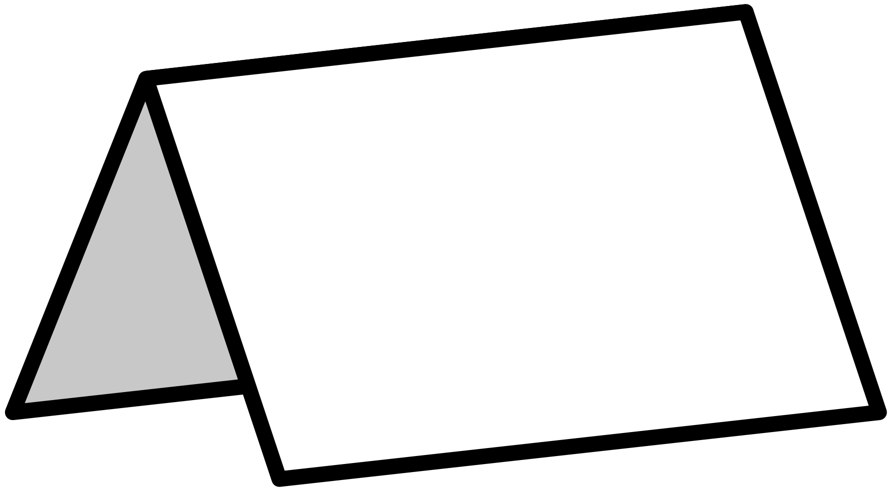
Why Quantum Computing?





https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

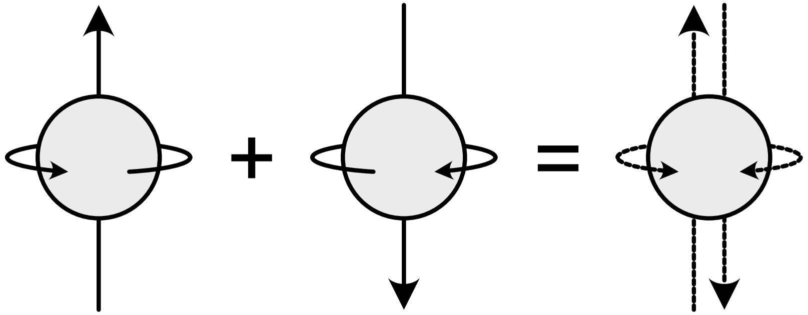
Why Quantum Computing?

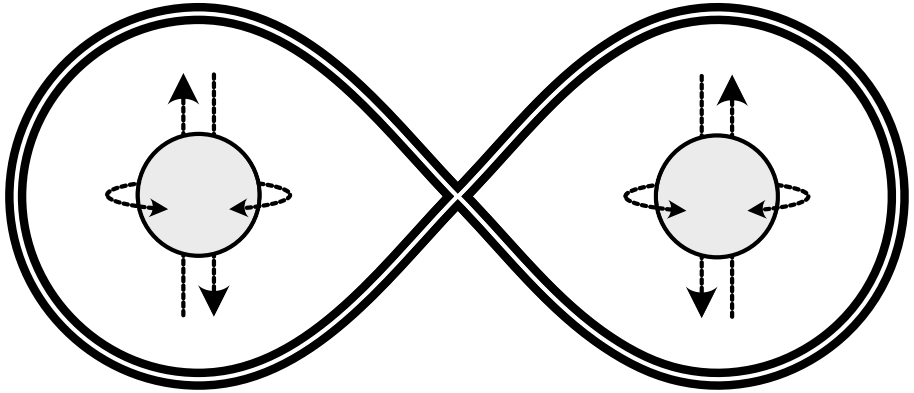


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

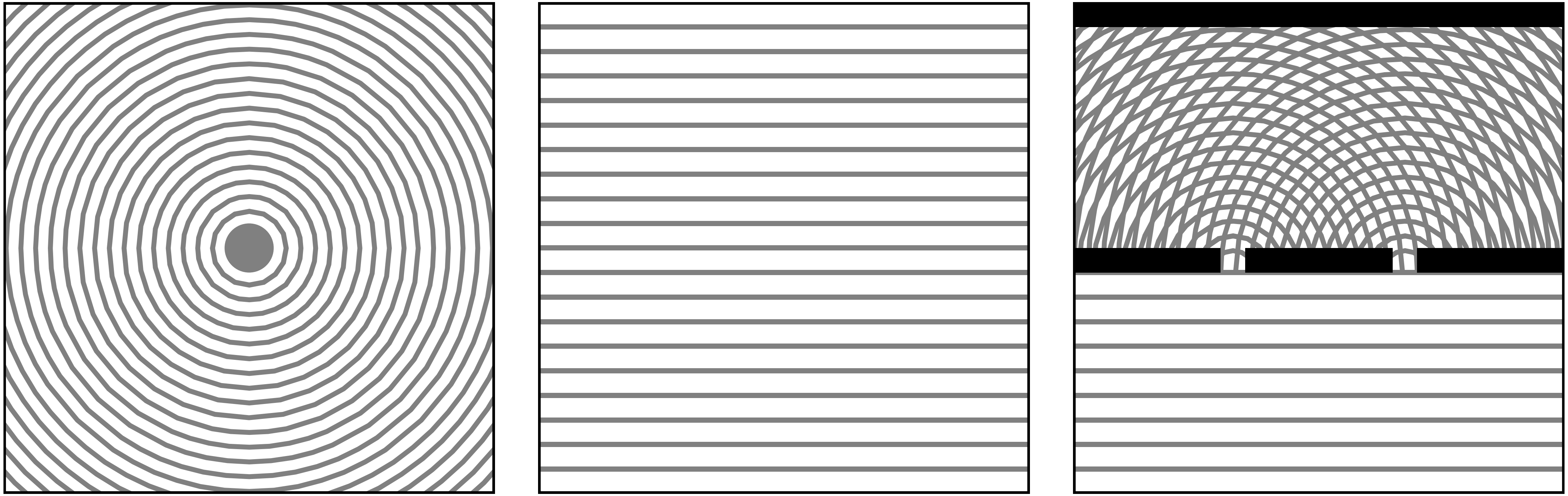
Superposition and Entanglement

- Quantum mechanics describes superposition and entanglement of quantum particles
- Quantum Computing can use these phenomenon to its advantage



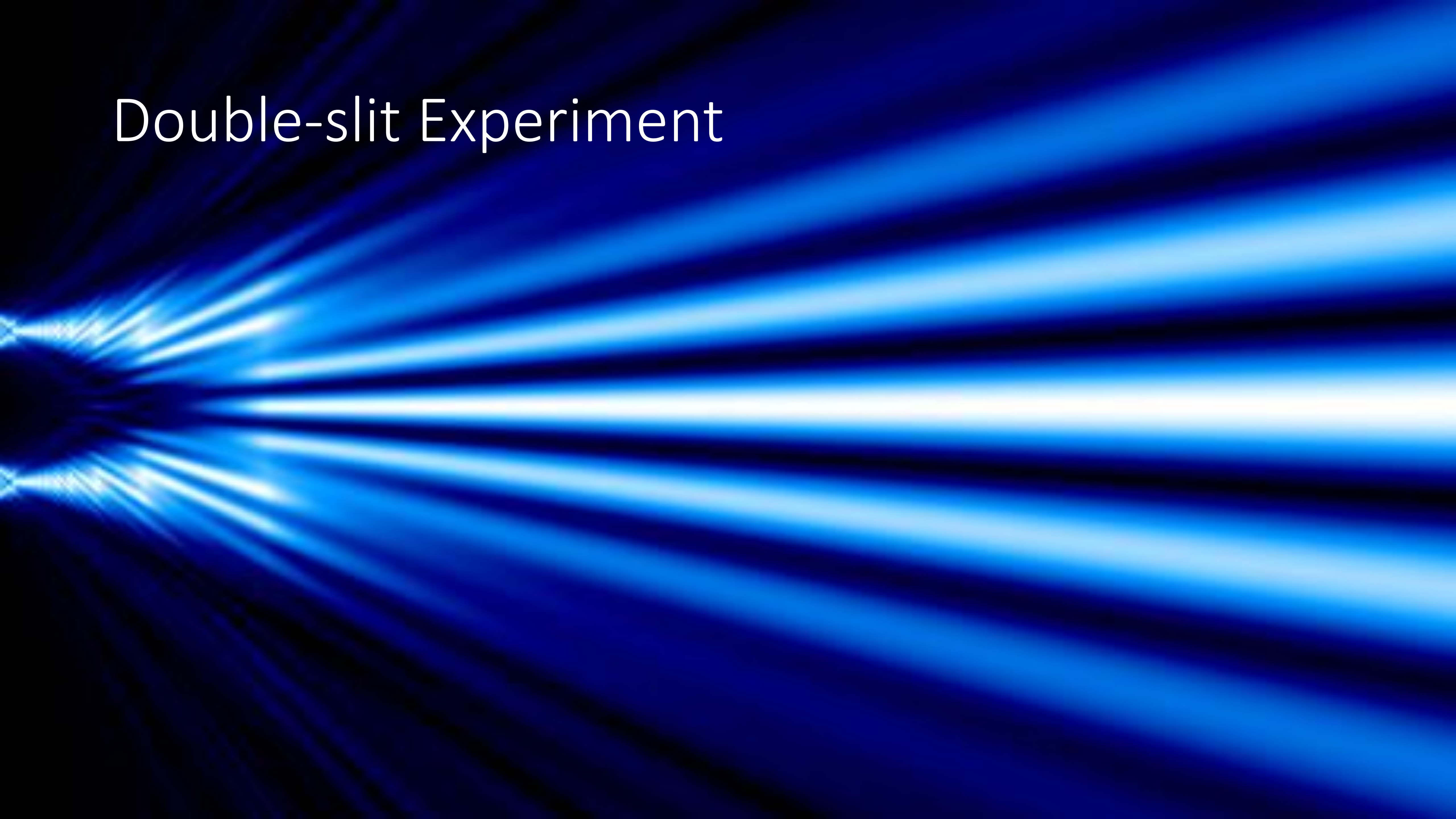


Double-slit Experiment

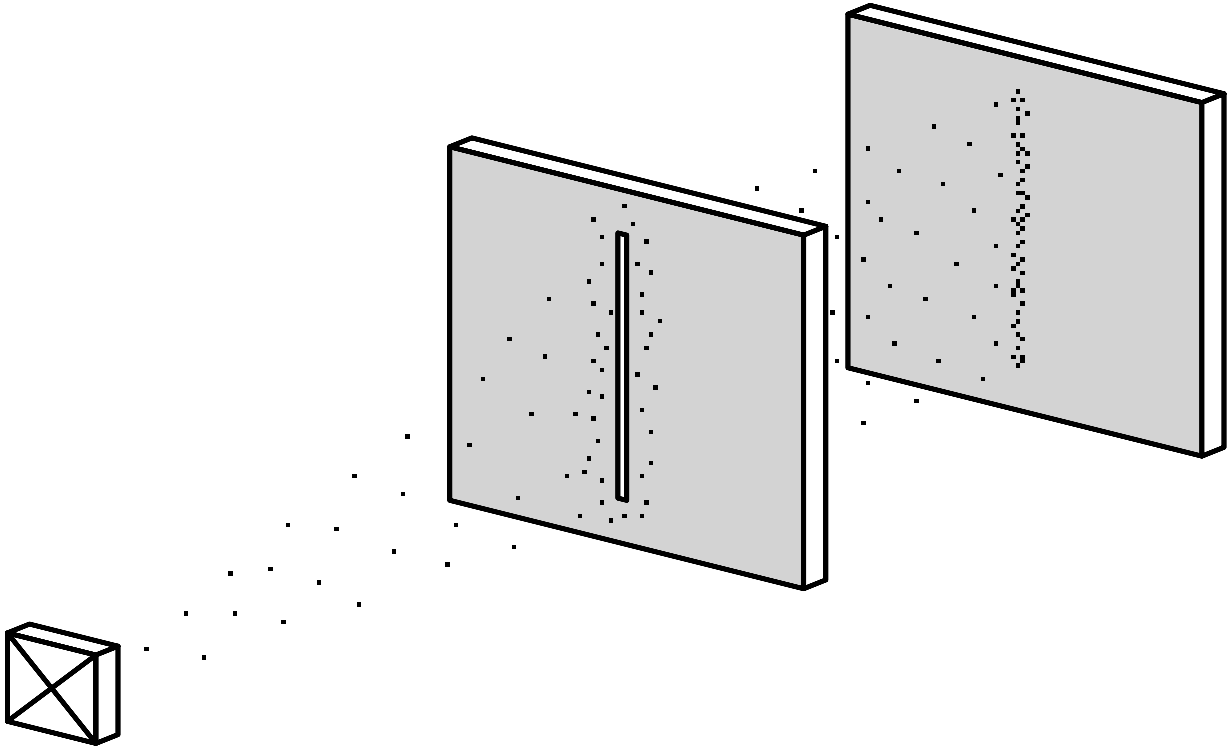


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Double-slit Experiment

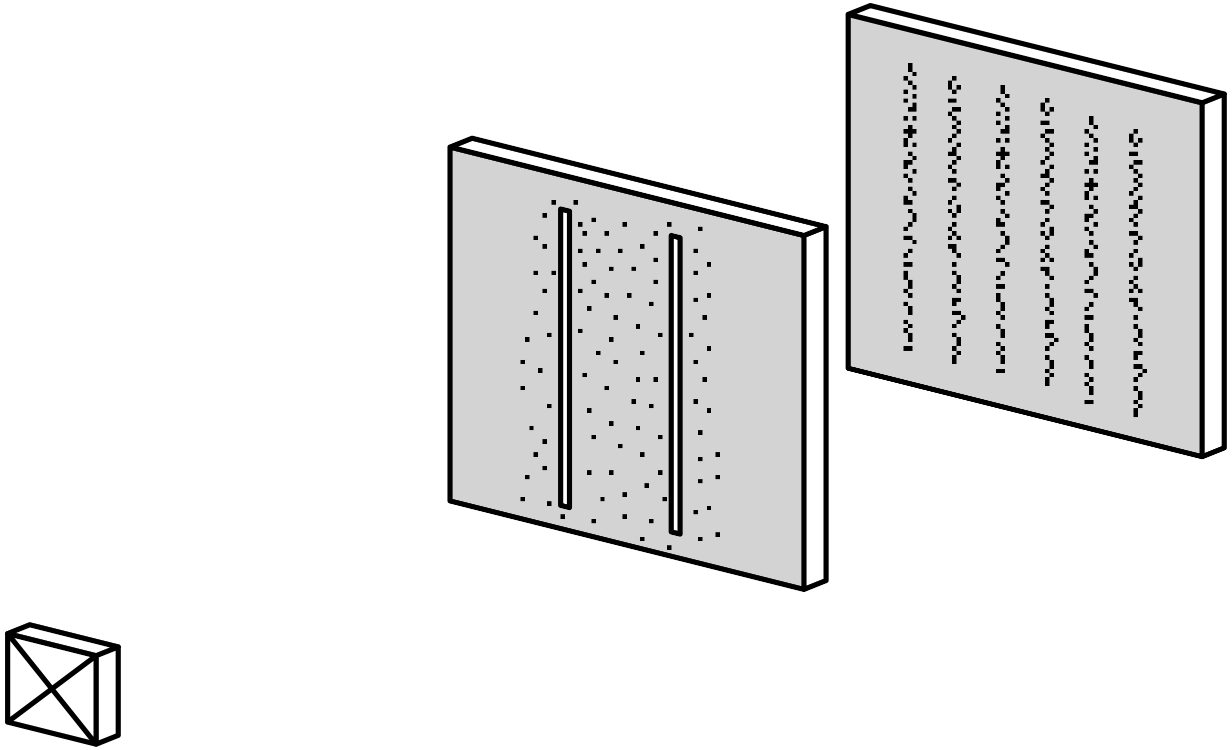


Double-slit Experiment



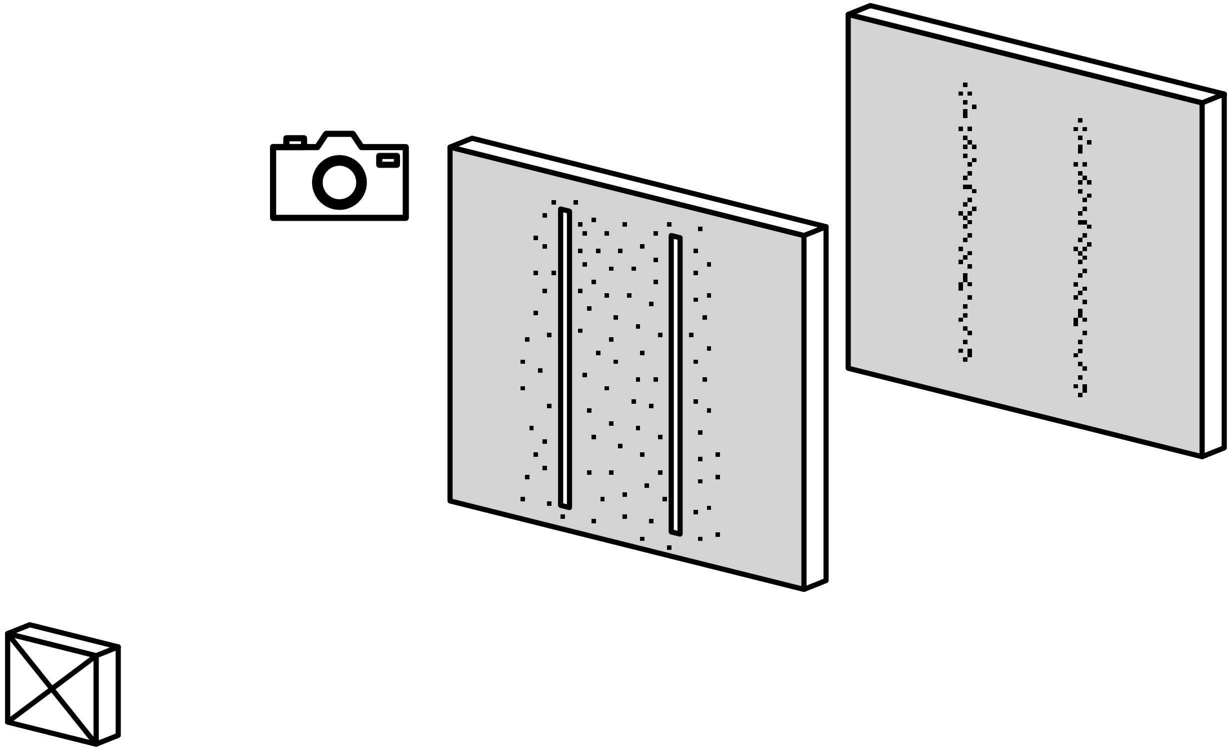
https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Double-slit Experiment



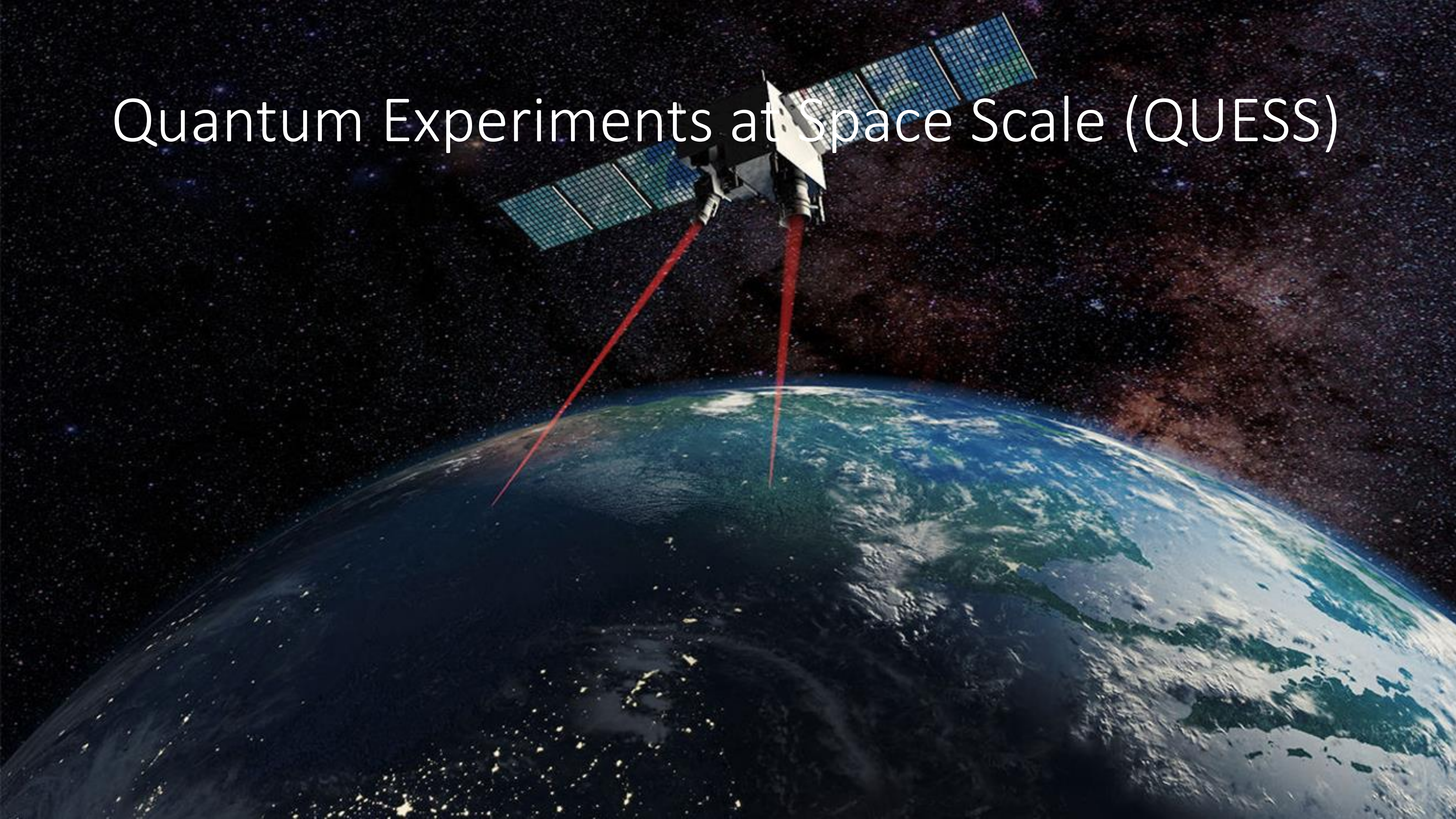
https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Double-slit Experiment



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum Experiments at Space Scale (QUESS)



Why Quantum Computing?

- Security
  - Public/private key encryption?
  - Could make current RSA encryption obsolete
  - QKD (Quantum Key Distribution)

$$3.167 \times 6.301 = 19.955.267$$

Why Quantum Computing?

- Drug development
  - It takes a quantum system to simulate a quantum system
  - Interactions between molecules
  - Gene sequencing
  - Protein folding

Why Quantum Computing?

- Machine Learning
  - Analyze large quantities of data
  - Fast feedback
  - Emulate human mind

?


CAN IT RUN CRYSIS?

Bits vs. Qubits

0

1

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Bits vs. Qubits

100110

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Bits vs. Qubits

$$|0\rangle \qquad\qquad |1\rangle$$

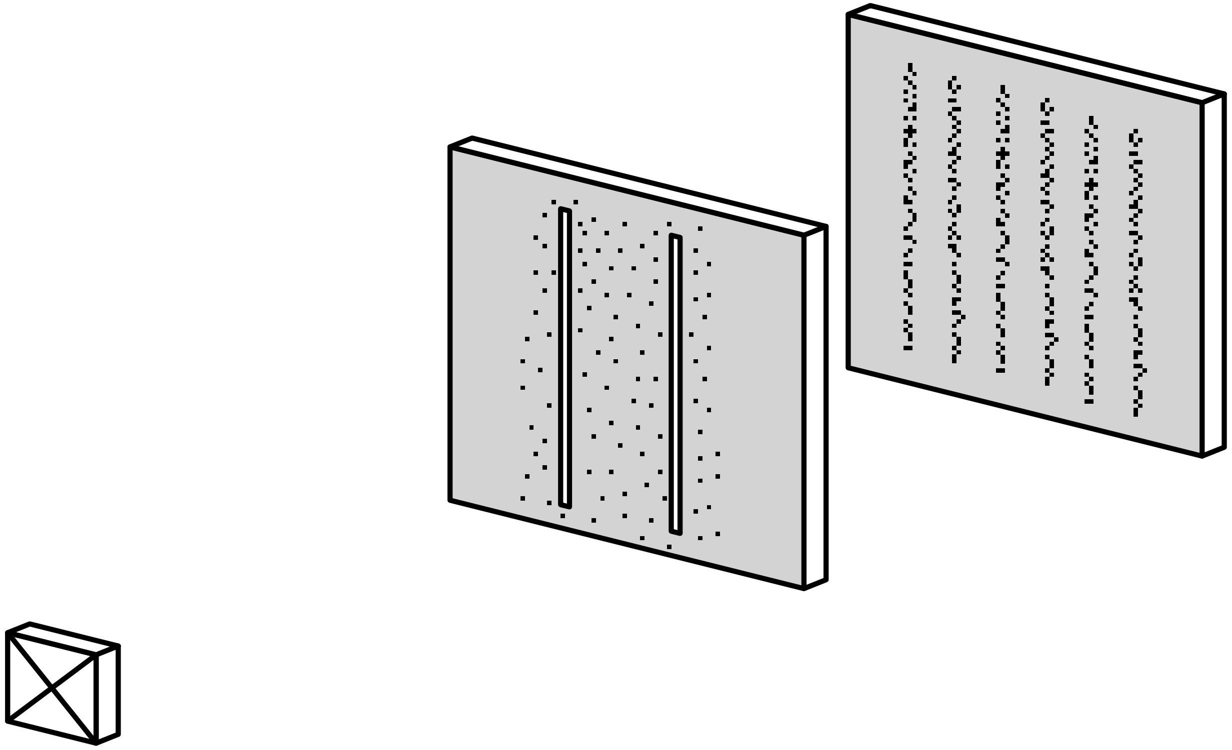
https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Bits vs. Qubits

$$|100110\rangle$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum state



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum state

$$\boldsymbol{\alpha}|0\rangle + \boldsymbol{\beta}|1\rangle$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum state

$$\boldsymbol{\alpha}|0\rangle + \boldsymbol{\beta}|1\rangle$$

$$|\boldsymbol{\alpha}|^2 + |\boldsymbol{\beta}|^2 = 1$$

Quantum state

$$\boldsymbol{\alpha}|0\rangle + \boldsymbol{\beta}|1\rangle$$

$$|\boldsymbol{\alpha}|^2 + |\boldsymbol{\beta}|^2 = 1$$

$$\boldsymbol{\alpha} = a + bi$$

$$\boldsymbol{\beta} = c + di$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum state

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

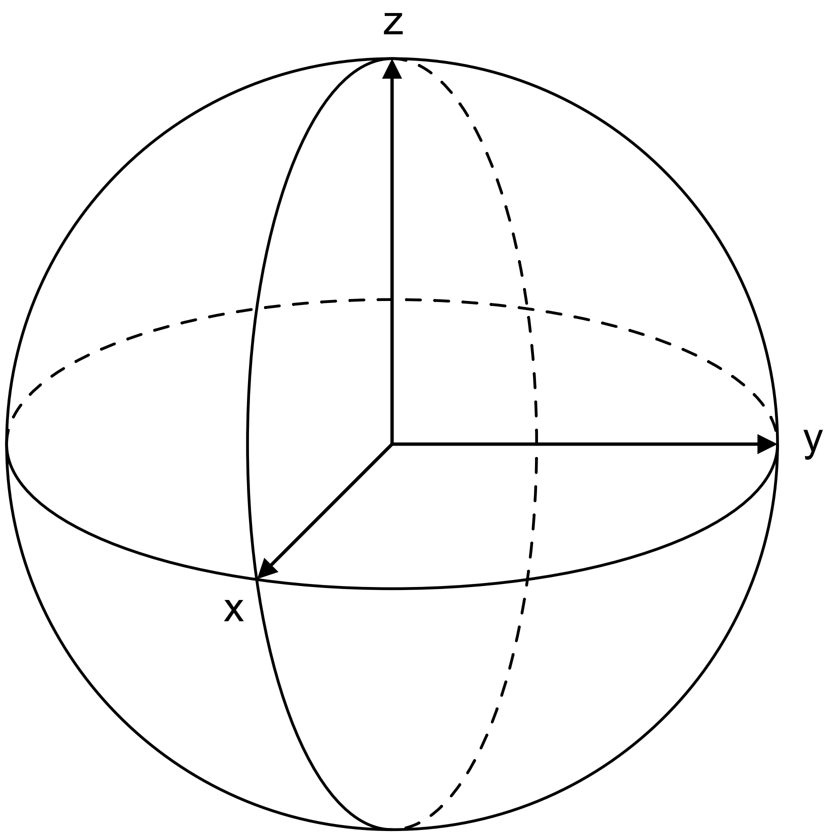
Quantum state

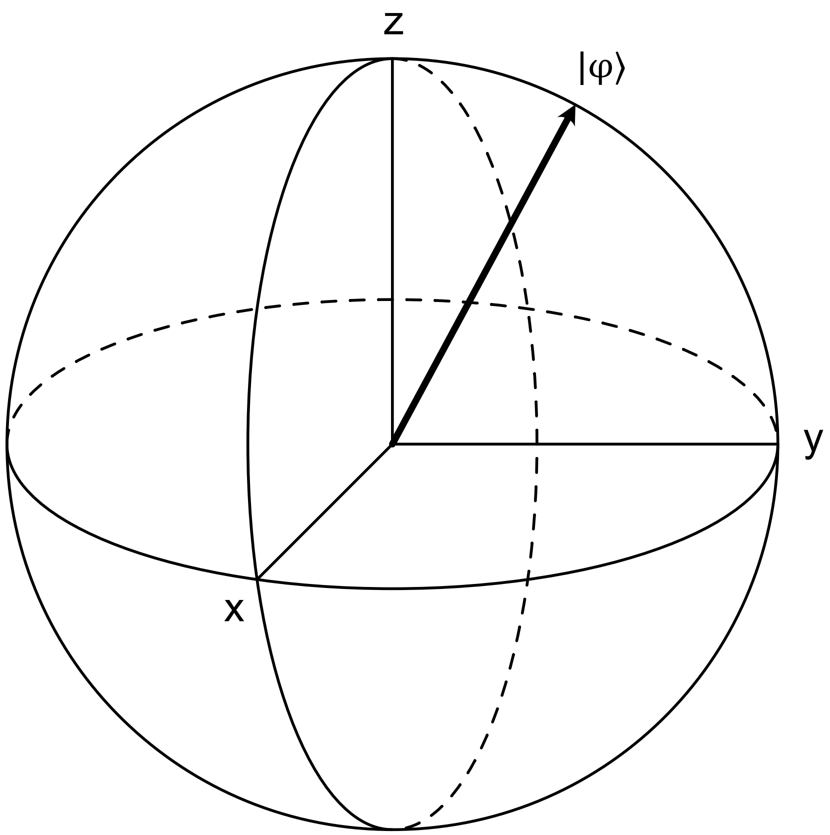
$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

Quantum state

- Classical bit 0, Quantum bit $|0\rangle$
- Classical bit 1, Quantum bit $|1\rangle$
- Quantum bit in superposition
- $\boldsymbol{\alpha}|0\rangle + \boldsymbol{\beta}|1\rangle$ where $|\boldsymbol{\alpha}|^2 + |\boldsymbol{\beta}|^2 = 1$
- $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are complex numbers $(\mathrm{a}i + \mathrm{b})$
- Value known after measurement
- Collapses to $|0\rangle$ with probability $|\boldsymbol{\alpha}|^2$ or $|1\rangle$ with probability $|\boldsymbol{\beta}|^2$

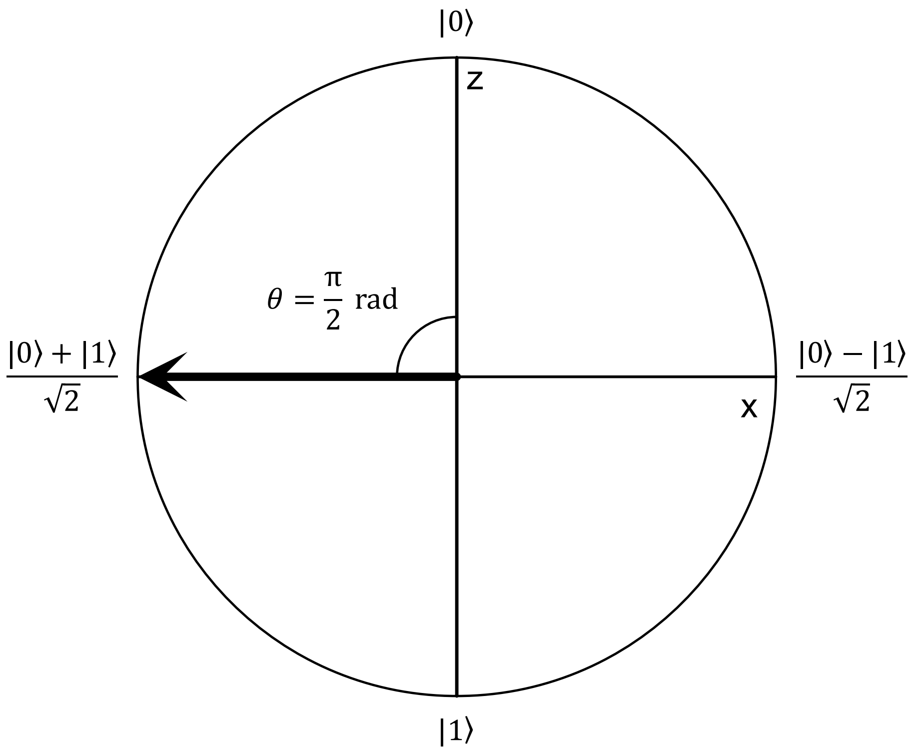
Quantum state

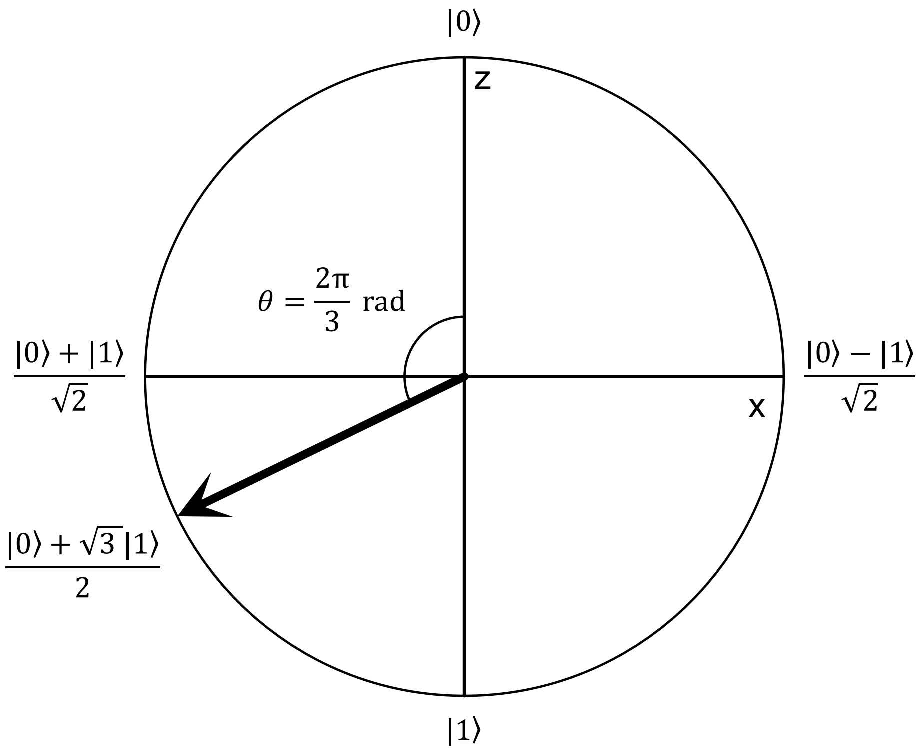




https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum state





https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum state

- 2 Qubit system (4 probabilities):

$$\boldsymbol{\alpha}|00\rangle + \boldsymbol{\beta}|01\rangle + \boldsymbol{\gamma}|10\rangle + \boldsymbol{\delta}|11\rangle$$

Quantum state

- 2 Qubit system (4 probabilities):

$$\boldsymbol{\alpha}|00\rangle + \boldsymbol{\beta}|01\rangle + \boldsymbol{\gamma}|10\rangle + \boldsymbol{\delta}|11\rangle$$

- 3 Qubit system (8 probabilities):

$$\boldsymbol{\alpha}|000\rangle + \boldsymbol{\beta}|001\rangle + \boldsymbol{\gamma}|010\rangle + \boldsymbol{\delta}|011\rangle + \boldsymbol{\varepsilon}|100\rangle + \boldsymbol{\epsilon}|110\rangle + \boldsymbol{\zeta}|101\rangle + \boldsymbol{\eta}|111\rangle$$

- 4 Qubit system (16 probabilities):

…

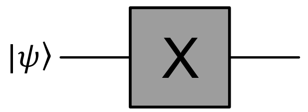
Quantum state

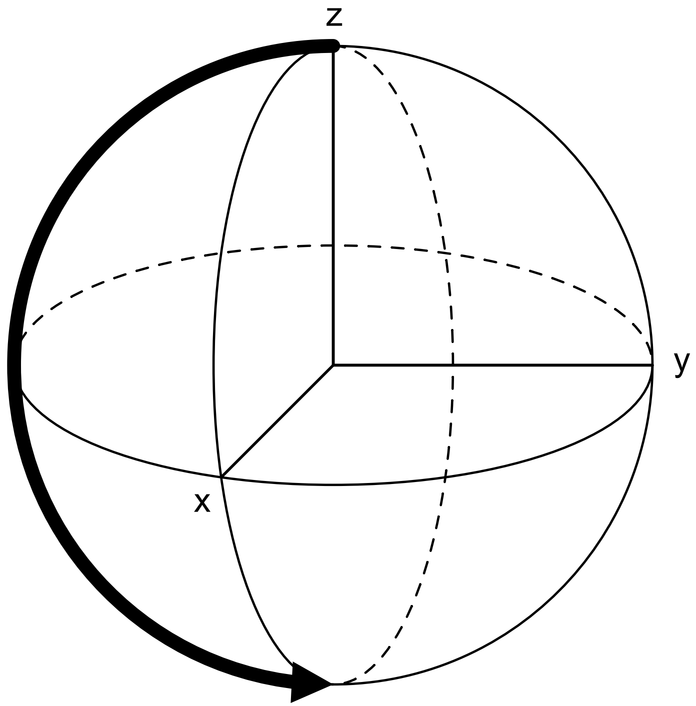
$$|0\rangle \qquad |1\rangle$$

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

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

X-gate

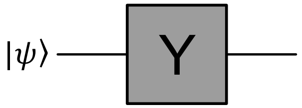


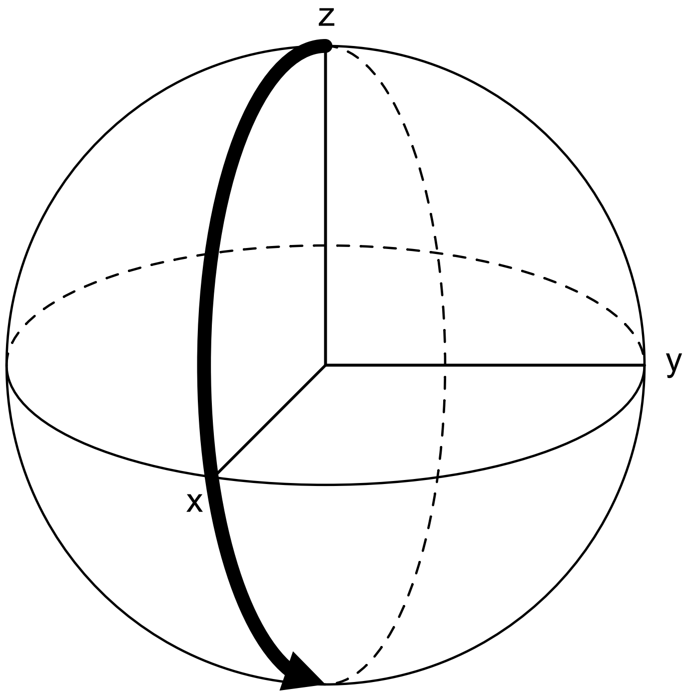


$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

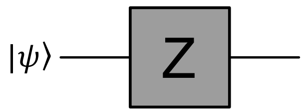
Y-gate

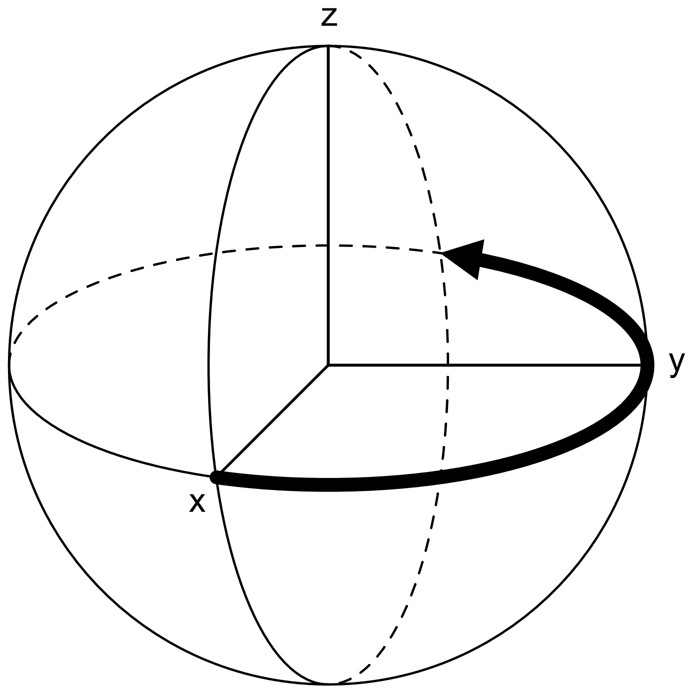




$$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Z-gate

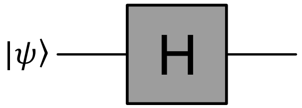


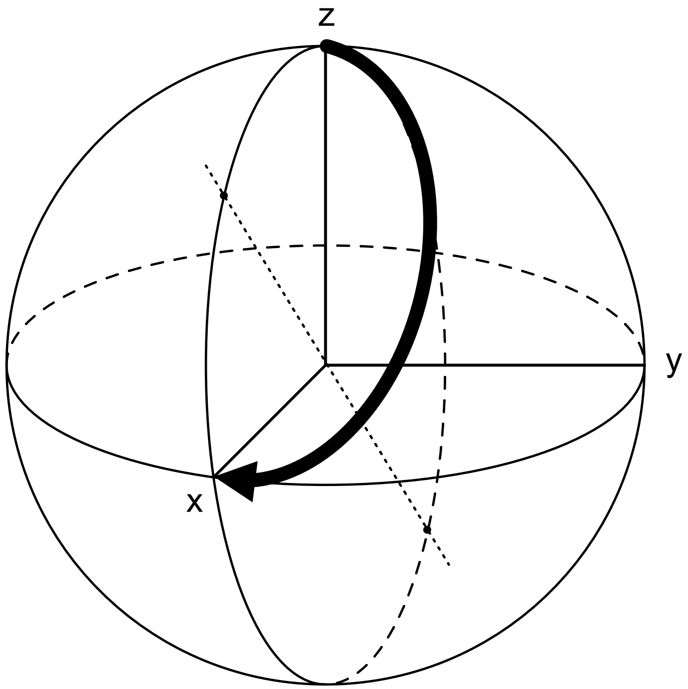


$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

H-gate

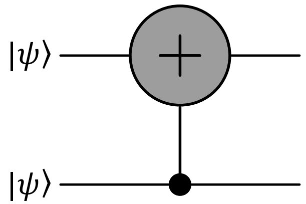




$$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix}$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

CNOT-gate



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

IBM Q Experience

https://quantum-computing.ibm.com

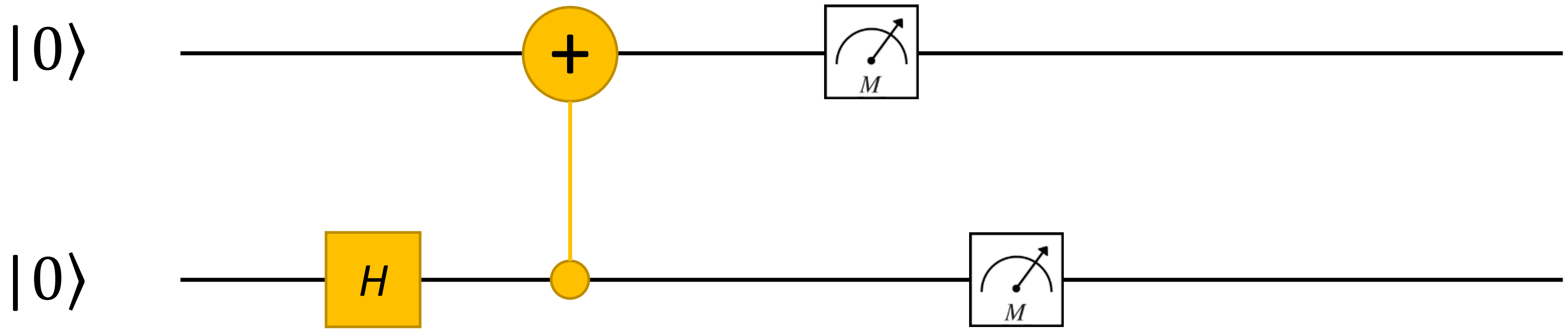
Microsoft Q#

https://www.microsoft.com/en-us/quantum/development-kit

Azure Quantum

- Quantum in the cloud
  - Optimization
  - Machine Learning
  - Quantum Simulation
- Access to quantum hardware
  - Microsoft (Topological)
  - IonQ & Honeywell (Ion Traps)
  - QCI (Superconducting)
- Q# & QDK
  - Quantum Intermediate Representation (QIR)

Entanglement



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

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ 0\\ \frac{1}{\sqrt{2}}\\ 0\end{pmatrix} \xrightarrow{\boldsymbol{CNOT}} \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}\begin{pmatrix}\frac{1}{\sqrt{2}}\\ 0\\ \frac{1}{\sqrt{2}}\\ 0\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ 0\\ 0\\ \frac{1}{\sqrt{2}}\end{pmatrix} = ?$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

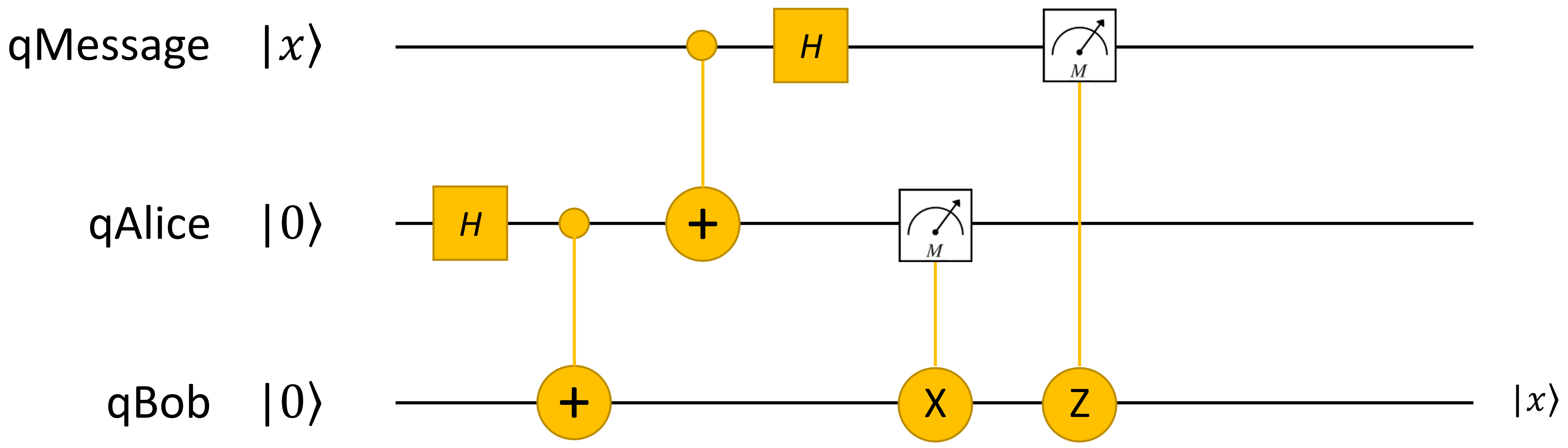
Entanglement

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

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \binom{a}{b} \otimes \binom{c}{d} \rightarrow \begin{array}{c} ac = \frac{1}{\sqrt{2}} \\ ad = 0 \\ bc = 0 \\ bd = \frac{1}{\sqrt{2}} \end{array}$$

This set of two qubits has a 50% chance of collapsing to $|00\rangle$ and a 50% chance of collapsing to $|11\rangle$

Teleportation



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Quantum Algorithms

- Deutch (1985)
  - Is there a problem that a Quantum Computer can solve faster than a Classical Computer?
  - Deterministic!

Quantum Algorithms

- Deutsch–Jozsa (1992)
  - Based on Deutch (for 1 bit), but applicable for n-bits
  - Deterministic!

Quantum Algorithms

- Grover's algorithm (1996)
  - “Searching a database”
  - Probabilistic!

Quantum Algorithms

- Shor's algorithm (1994)
  - Prime factorization of integers
  - Combination of classical and quantum algorithm
  - Probabilistic!

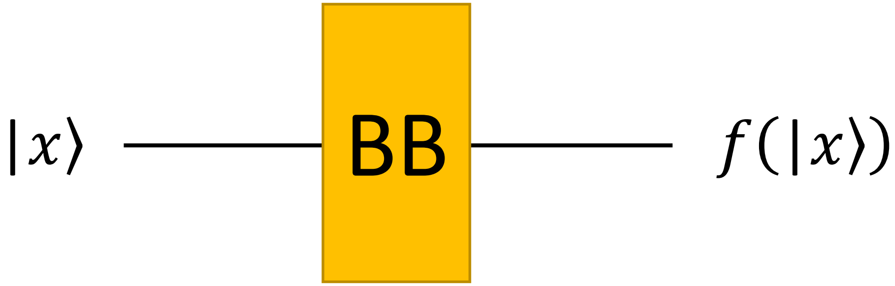
Deutsch's algorithm

Can a Quantum Computer be quicker than a Classical Computer?

A Black-Box containing a function on one bit

How many operations do you need to figure out the function if input and output is know?

- On a Classical Computer?
- On a Quantum Computer?



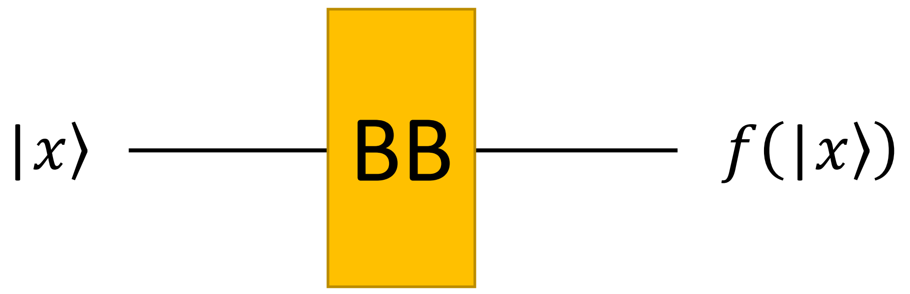
Deutsch's algorithm

It is important to ask the right question!

A Black-Box containing a function on one bit

How many operations do you need to figure out if the function is CONSTANT or VARIABLE if input and output is know?

- On a Classical Computer?
- On a Quantum Computer?



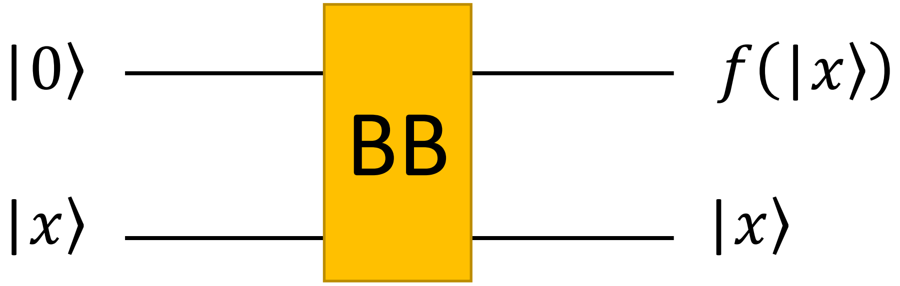
Deutsch's algorithm

It is important to ask the right question!

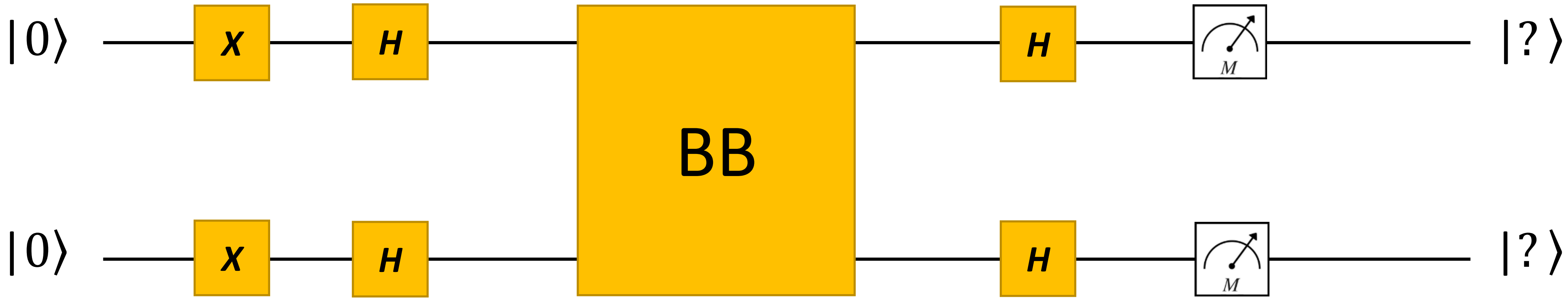
A Black-Box containing a function on one bit

How many operations do you need to figure out if the function is CONSTANT or VARIABLE if input and output is know?

- On a Classical Computer?
- On a Quantum Computer?



Deutsch's algorithm



If BB is a constant function → Quantum state will always measure to $|11\rangle$

If BB is a variable function → Quantum state will always measure to $|01\rangle$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

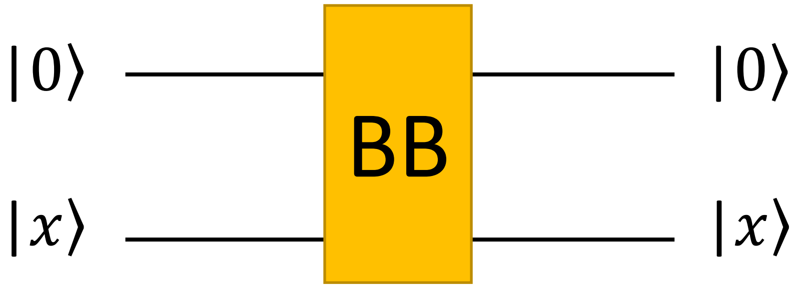
Deutsch's algorithm

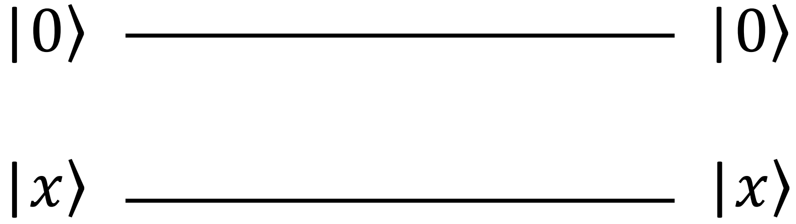
Constant-0

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Constant-0

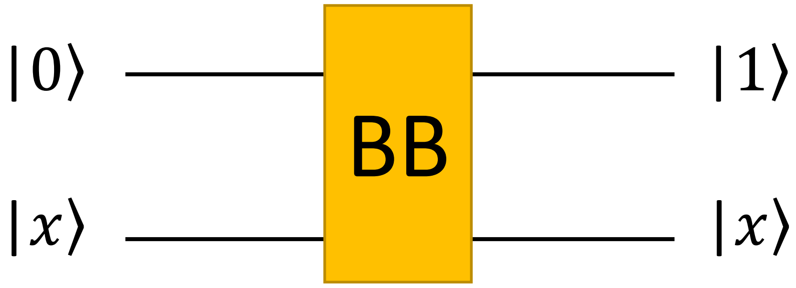


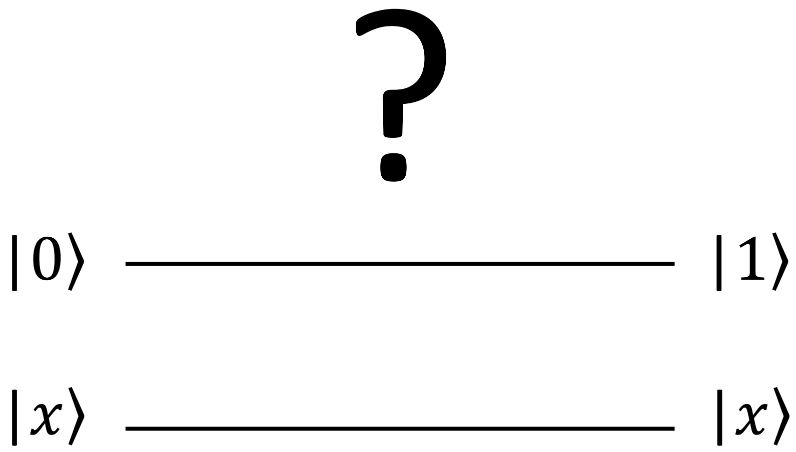


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Constant-1

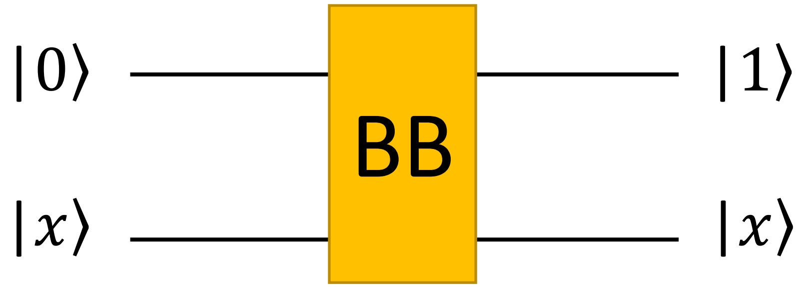


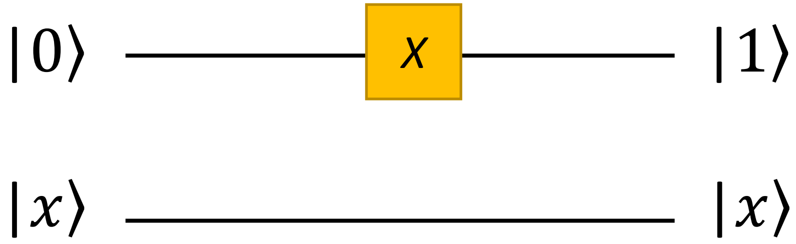


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Constant-1

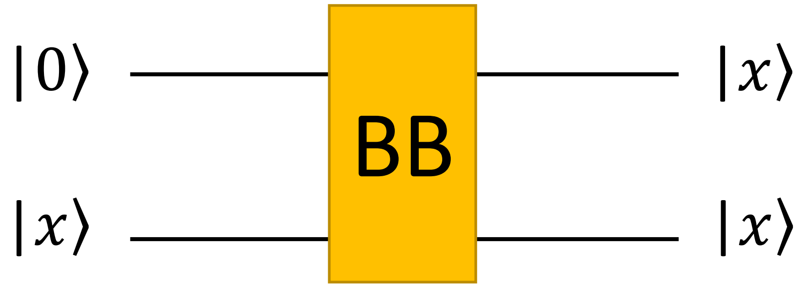


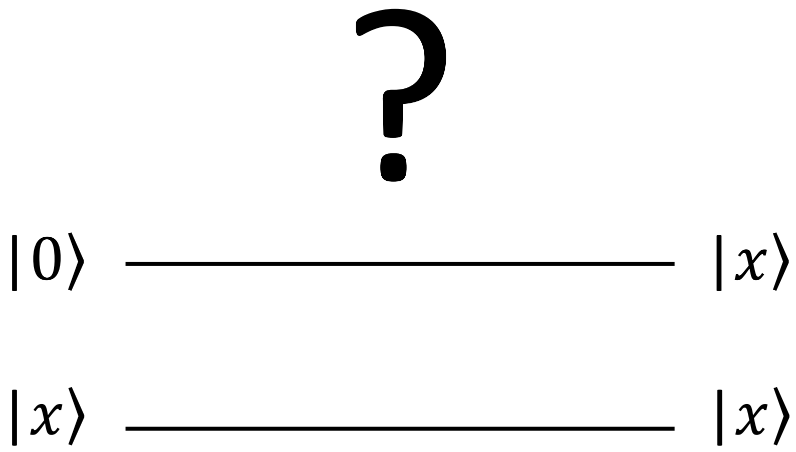


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Identity

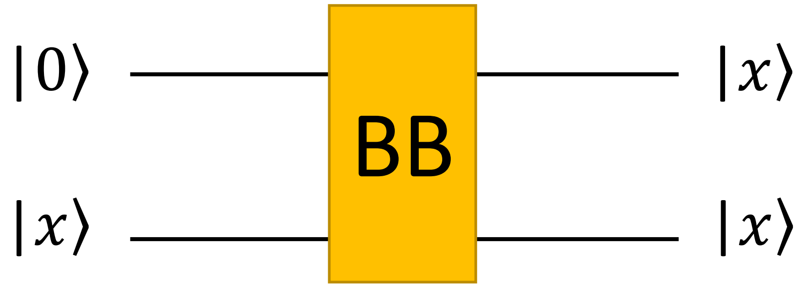


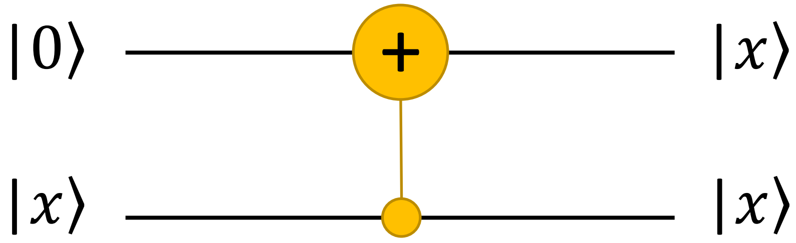


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Identity

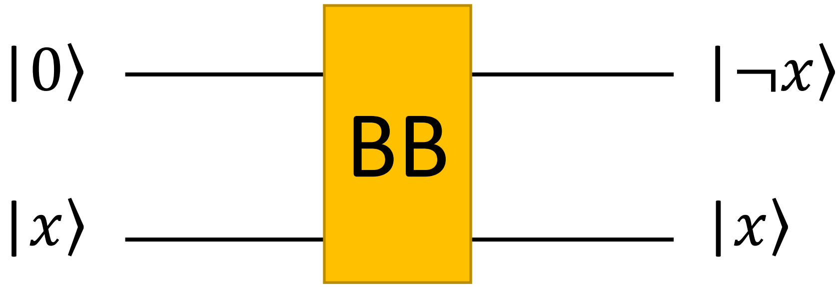


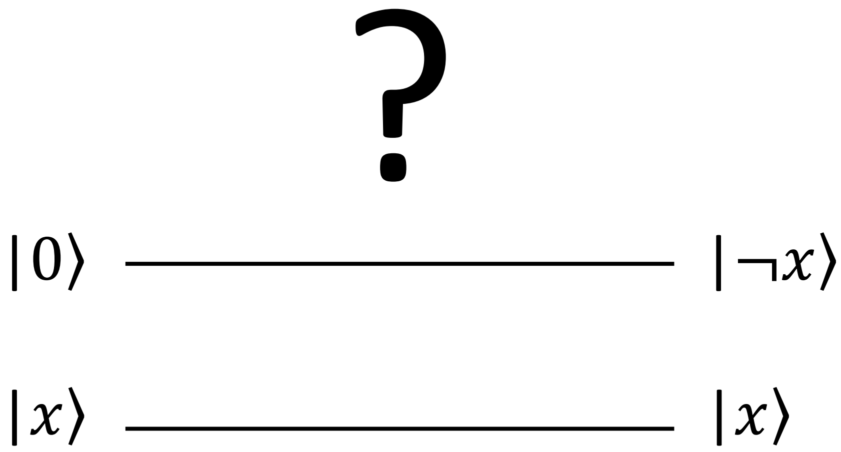


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Negation

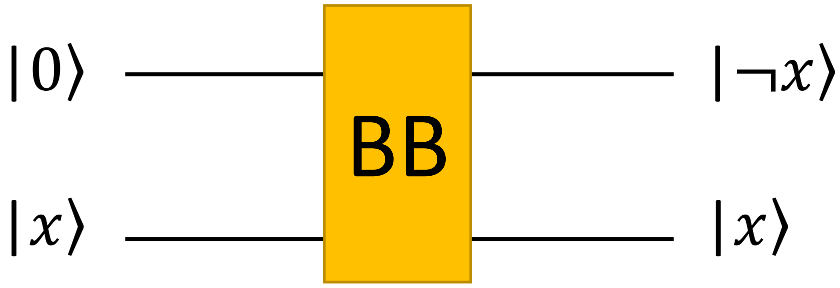


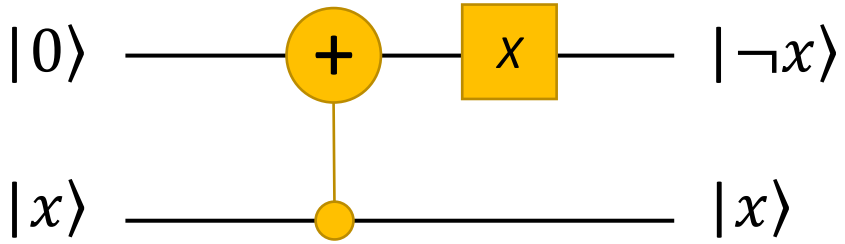


https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Negation

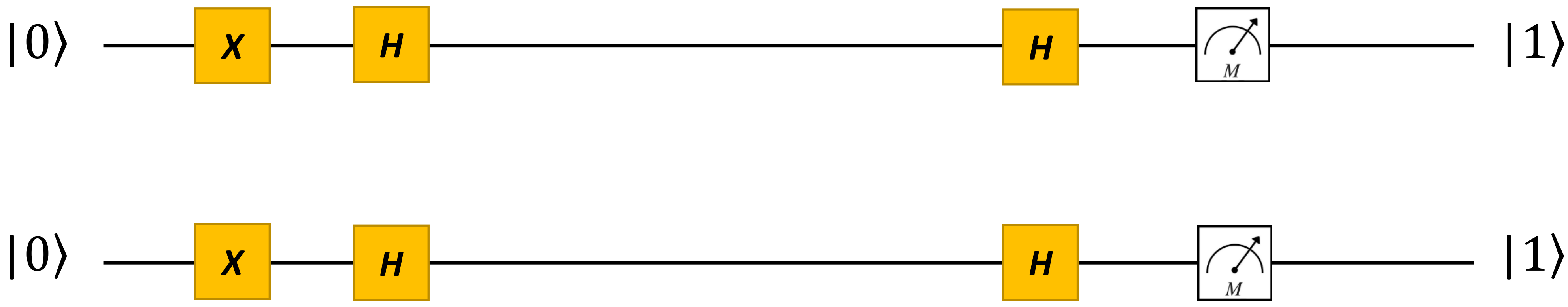




https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Constant-0 (circuit overview)



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Constant-0 (calculated proof – part 1)

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}$$

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}$$

Deutsch's algorithm

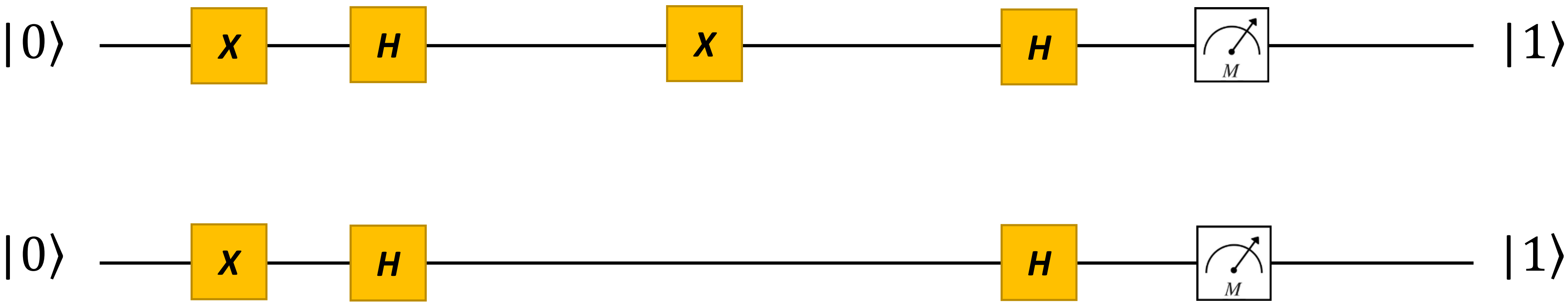
Constant-0 (calculated proof – part 2)

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

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

Deutsch's algorithm

Constant-1 (circuit overview)



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Constant-1 (calculated proof – part 1)

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}$$

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}$$

Deutsch's algorithm

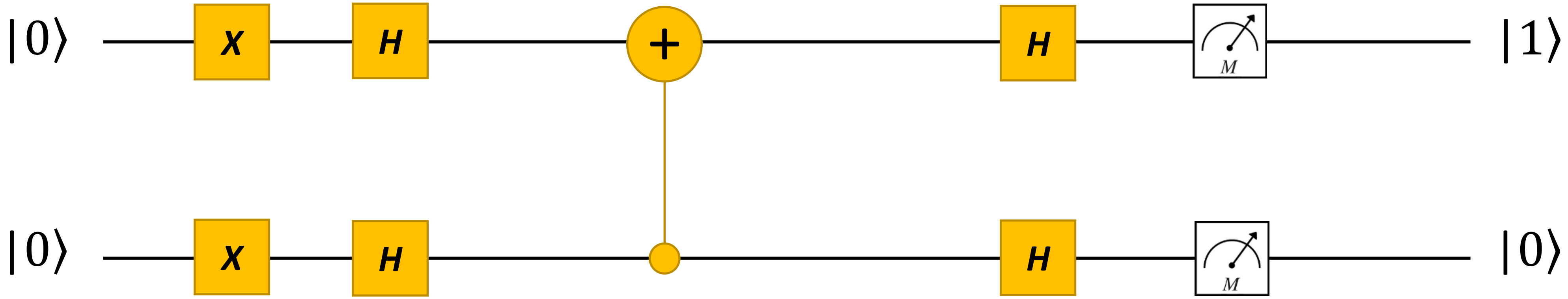
Constant-1 (calculated proof – part 2)

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} = |1\rangle$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{Id}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

Deutsch's algorithm

Identity (circuit overview)



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Identity (calculated proof – part 1)

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}$$

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}$$

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

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Identity (calculated proof – part 2)

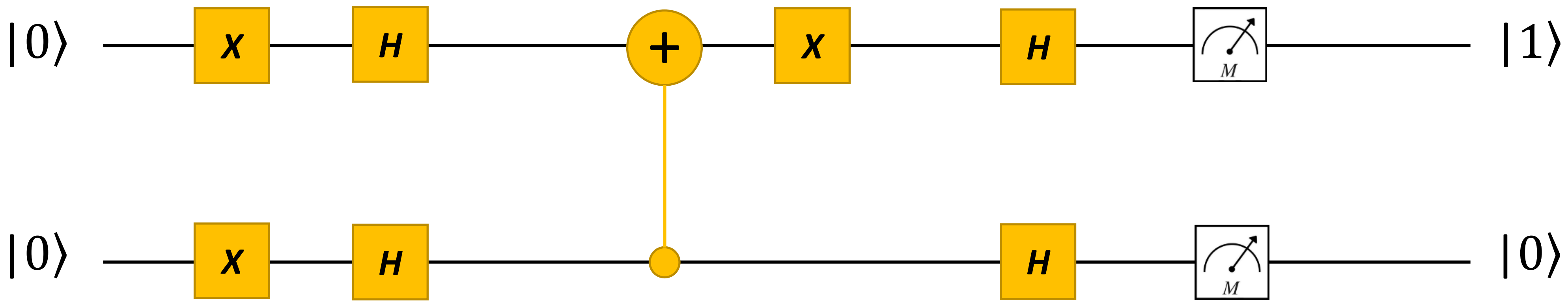
$$\begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \underset{\rightarrow}{\boldsymbol{CNOT}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} =$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \underset{\rightarrow}{\boldsymbol{H}} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \underset{\rightarrow}{\boldsymbol{H}} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

Deutsch's algorithm

Negation (circuit overview)



https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Negation (calculated proof – part 1)

$$
\left.
\begin{array}{l}
|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix} \\
|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix}0 & 1\\1 & 0\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix}
\end{array}
\right.
= \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix} \otimes \begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{-1}{\sqrt{2}}\end{pmatrix} = \begin{pmatrix}\frac{1}{2}\\ -\frac{1}{2}\\ -\frac{1}{2}\\ \frac{1}{2}\end{pmatrix}
$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020

Deutsch's algorithm

Negation (calculated proof – part 2)

$$\begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \xrightarrow{\boldsymbol{CNOT}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} =$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{X}} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{-1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} = |1\rangle$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{Id}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \xrightarrow{\boldsymbol{H}} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

https://github.com/Djohnnie/QSharp-and-AzureQuantum-CanaryWharf-2020