IBM Quantum

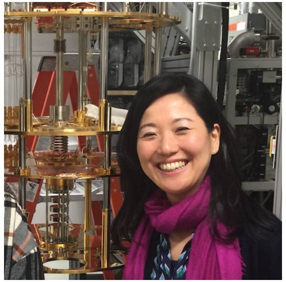
第2回 Qiskitハンズオン研修　量子アルゴリズム編
Quantum Algorithms

IBM東京基礎研究所
量子コンピューティング　量子開発者コミュニティー
小林 有里

IBM

自己紹介

【講師】　小林有里　（こばやし・ゆり）
日本アイ・ビー・エム株式会社
東京基礎研究所 量子コンピューティング
Quantum Developer Community担当



【経歴】
- 物理学科 固体物理 有機超伝導体 材料研究
- 日本IBM（株）ソフトウェア組込開発事業に配属。ストレージ製品の開発に従事。
- 2014年に日本IBM 東京基礎研究所に入所。
- ワークプレイス・アクセシビリティをテーマにAIを活用した障害者や高齢者の就労支援技術の研究およびソリューション開発に従事。
- 2019年より量子コンピューティングチームにて、Qiskitの開発者向けコミュニティを担当。Qiskitの研修、大学での授業展開をはじめ、量子コンピューターを使ったプログラミングコンテスト、ハッカソンの主催などを通じて、量子ネイティブと呼ばれる量子コンピューターをつかいこなせる世代の育成に注力。

当研修について

- **前回**
- IBMの量子コンピューターについて
- 量子計算の基礎
- Qiskitをつかった量子回路の作成(ハンズオン)

- **本日**
- 量子アルゴリズム編
  - ドイチ・ジョザ アルゴリズム
  - バーンスタイン・ヴァジラニアルゴリズム
  - Qiskitをつかった実装

※進み具合、興味関心に応じて内容を調整させていただく可能性があります。

1. ドイチ・ジョザアルゴリズム

ドイチ・ジョザアルゴリズムの意義

- ブラックボックス問題を従来のコンピューターよりも少ない問合せ量で解くことができることを証明した最初の量子アルゴリズムのひとつ。
- オラクル、位相キックバックなど、多くの量子アルゴリズムで使われるテクニックを有している。
- サイモンのアルゴリズム、ショアのアルゴリズムのインスピレーションとなった。

ブラックボックス問題 (Black Box Problem)：問合せに対しては特定のnビットの出力を返す関数。関数の中身は問合せする側には隠されていることから、Blackbox関数とも呼ばれる

問合せ量 (Query Complexity)：解を得るために照会をかける回数のこと

52枚のトランプ

相手が52枚入りのトランプ2箱を用意し、隠れたところで以下のうちどちらかを準備したとする

1. 赤だけ(または黒だけ)で52枚　(一定)
2. 赤と黒半々(26枚ずつ)で52枚　(均等)

あなたは相手が準備した52枚のトランプが**一定、または均等のどちらか**を判定しなければならない

何回ひけば相手のトランプが一定または均等かがわかるか？

- 古典：ベスト 2回ひいて判定可　ワースト 27枚ひいて判定可　(1+52/2)
- 量子コンピューター：1回で判定可

ドイチ・ジョザ問題

オラクル：判定を行う必要があるときに登場する

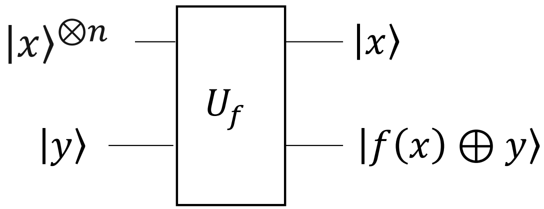
ここでは2^nバイナリ文字列を入力として、0または1を返す関数$f(x)$

$$f(x): \{0,1\}^n \rightarrow \{0,1\}$$

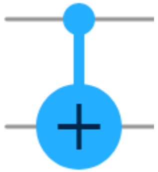
ドイチ・ジョザで使われるテクニック

量子オラクル：制御ユニタリU_f

$$U_f: |x\rangle|y\rangle \mapsto |x\rangle|f(x) \oplus y\rangle$$

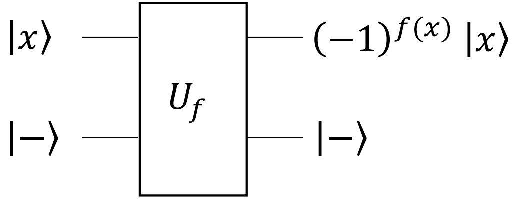


CNOTは有名な制御ユニタリ



位相キックバック

$$U_f: |x\rangle|-\rangle \mapsto (-1)^{f(x)}|x\rangle|-\rangle$$



本来ターゲット側に
エンコードされる位相情報が
制御側に反映　∴キックバック

デルフォイのオラクル（神託）

- デルフォイは、古代ギリシャ中部にあった地名。
  オラクル（ 神託）とは神様からのお告げ・予言の意。
  デルフォイにあったアポロン神殿に都市国家(ポリス)が
  参じて政治、外交、戦略についてオラクル（神託）に
  うかがいをたてていた。
- アポロン神殿に仕える巫女たちも、問合せに対して
  「はい」「いいえ」で託宣をくだしていたという。
- 計算機科学においては、計算複雑性理論や計算可能性
  理論における抽象機械の一種であり、決定問題の研究
  で使われる。

画像：Wikimedia Commons

オラクルの役割

オラクルを関数をfとして考えると、以下のように表現できます。

f: $\{0,1\}^{n} \rightarrow \{0,1\}$

古典表現

$$f(x) = \begin{cases} 1 & \text{トランプが黒のとき} \\ 0 & \text{それ以外} \end{cases}$$


量子表現：ケット記号を使います

$$f|\Psi\rangle = \begin{cases} |1\rangle & \text{トランプが黒のとき} \\ |0\rangle & \text{それ以外} \end{cases}$$


量子計算ではオラクルをユニタリ変換U_fとして捉え、また$|0\rangle \rightarrow |1\rangle$といった量子状態の変化をマイナス符号（−）で記述します。 ※物理では「−」を「位相変換」と呼んだりします

例えば2量子ビットの重ね合わせのうち、$|01\rangle$が探している条件を満たすとき

$$U_f|\Psi\rangle = \begin{cases} -|01\rangle & |\Psi\rangle = |01\rangle \text{のとき} \\ |\Psi\rangle & |\Psi\rangle = |00\rangle\ |10\rangle\ |11\rangle \text{のとき} \end{cases}$$

オラクル U_f は条件を満たすものを識別する分類機のような役割を果たしてくれる。

演習：オラクルをつくってみよう

前提：

$$c_{00}|00\rangle + c_{01}|01\rangle + c_{10}|10\rangle + c_{11}|11\rangle = \begin{bmatrix} c_{00} \\ c_{01} \\ c_{10} \\ c_{11} \end{bmatrix} \quad \frac{1}{\sqrt{2^2}}(|00\rangle + |01\rangle + |10\rangle + |11\rangle \quad \mathrm{H}^{\otimes n}|0\rangle = \frac{1}{\sqrt{2^n}}\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

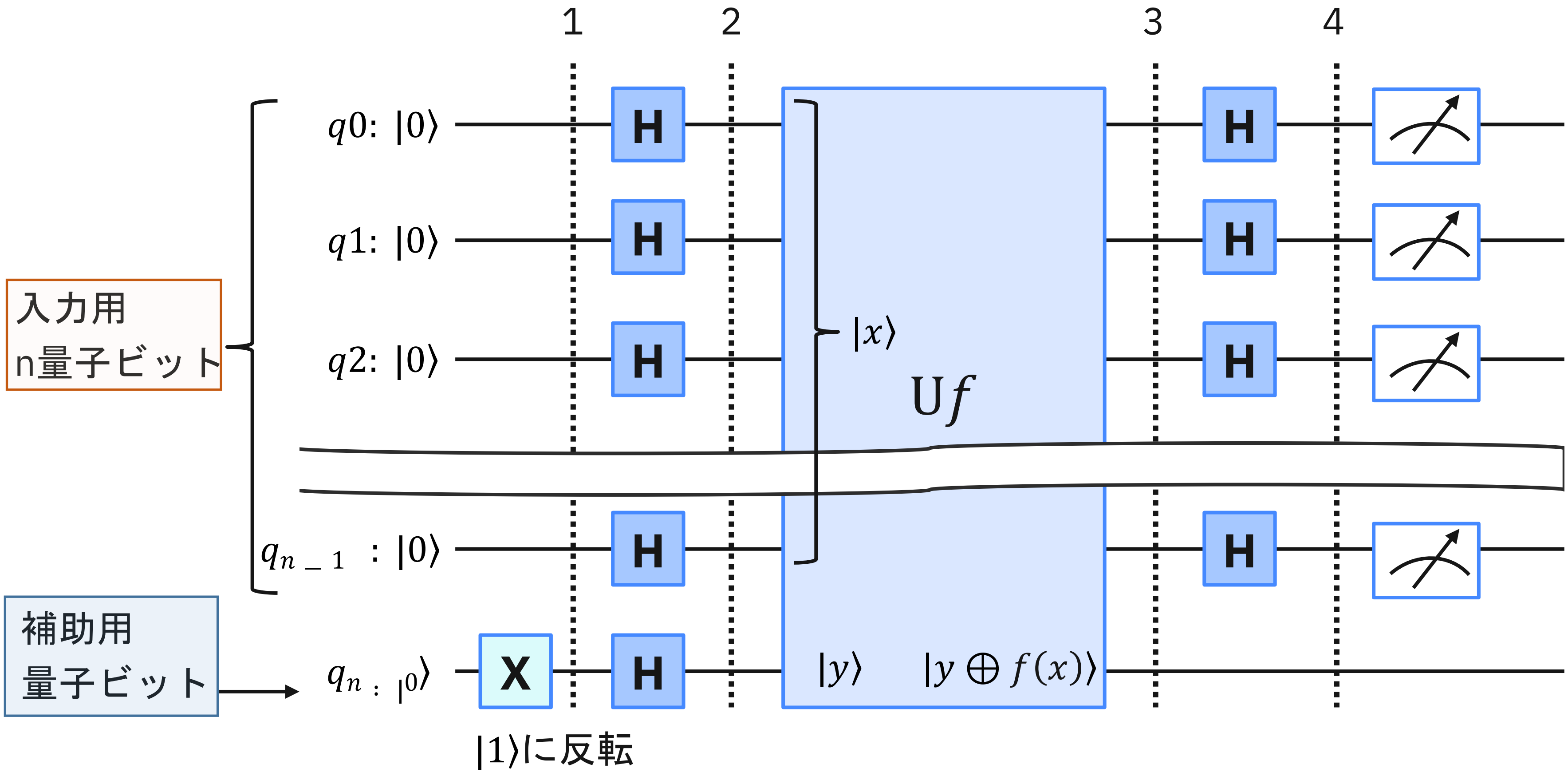
演習：$|\Psi\rangle = |01\rangle$のときに位相反転（$-|01\rangle$）をするオラクル$U_f$は4 x4の行列でかけます。
オラクル U_f はどのような行列になるか考えてみましょう。

$$U_f|\Psi\rangle = \begin{cases} -|01\rangle & |\Psi\rangle = |01\rangle \text{のとき} \\ |\Psi\rangle & |\Psi\rangle = |00\rangle\ |10\rangle\ |11\rangle \text{のとき} \end{cases}$$

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

ドイチ・ジョザのアルゴリズム

IBM **Quantum**



ドイチ・ジョザのアルゴリズム

IBM **Quantum**

Step1: 入力用n量子ビット$|0\rangle^{\otimes n}$,アンシラは$|1\rangle$に初期化

Step2: 重ね合わせをつくる　$H^{\otimes n}\,|0\rangle^{\otimes n} \otimes H|1\rangle$

$$=\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n}(|x\rangle|-\rangle)$$

モジュロ２加算
(2進数のときは**XOR**と同じ役割)

Step3: オラクルU_fを適用　$U_f\!: |x\rangle \otimes |y\rangle \mapsto |x\rangle \otimes |y \oplus f(x)\rangle$

入力n量子ビット　　アンシラ

$$\boxed{\left(\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n}|x\rangle\right)} \otimes \boxed{\frac{1}{\sqrt{2}}\left(\,|0 \oplus \mathrm{f(x)}\rangle\text{-}\,|1 \oplus \mathrm{f(x)}\rangle\right)}$$

$f(x)$=0: アンシラは$\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle$

$f(x)$=1: アンシラは$\frac{1}{\sqrt{2}}(|1\rangle - |0\rangle) = -|-\rangle$

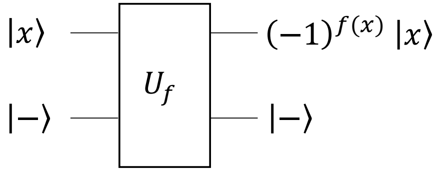
$$\boxed{(-1)^{f(x)}\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)}$$

つづく

位相キックバック

Step3: オラクルU_fを適用のつづき

位相キックバック：
位相が制御ビット側にエンコードされる



$$\boxed{\left(\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n}|x\rangle\right)} \otimes \boxed{\underline{(-1)^{f(x)}}\,\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)}$$

グローバル位相を外に括り出す

アンシラは測定しないのでこの先無視してよい

$$= \boxed{\left(\frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n}(-1)^{f(x)}|x\rangle\right)} \otimes \boxed{\left(\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right)}$$

一定関数：全て$f(x)$=0, or 全て$f(x)$=1

$(-1)^0|x\rangle$ $(-1)^1|x\rangle$

$$H^{\otimes n}\begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix} = \frac{1}{\sqrt{2^n}}\begin{bmatrix}1\\1\\1\\\vdots\\1\end{bmatrix} \xrightarrow{U_f} \frac{1}{\sqrt{2^n}}\begin{bmatrix}1\\1\\1\\\vdots\\1\end{bmatrix} \text{ or } \frac{1}{\sqrt{2^n}}\begin{bmatrix}-1\\-1\\-1\\\vdots\\-1\end{bmatrix}$$

均等のとき半分が $f(x)$=1, 半分が$f(x)$=0

半分$(-1)^0|x\rangle$ 半分$(-1)^1|x\rangle$

$$H^{\otimes n}\begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix} = \frac{1}{\sqrt{2^n}}\begin{bmatrix}1\\1\\1\\\vdots\\1\end{bmatrix} \xrightarrow{U_f} \frac{1}{\sqrt{2^n}}\begin{bmatrix}-1\\1\\1\\\vdots\\-1\end{bmatrix}$$

ドイチ・ジョザのアルゴリズム

Step4:Hをかけてから測定する

一定関数

全て$f(x)$=0のケース　　(Recall H |+⟩ → |0⟩)

$$H^{\otimes n}\frac{1}{\sqrt{2^n}}\begin{bmatrix}1\\1\\1\\\vdots\\1\end{bmatrix}\longrightarrow\begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix}$$

一定のときは
100%の確率で
$|000\ldots0\rangle$ を計測

全て$f(x)$=1のケース

$$H^{\otimes n}\frac{1}{\sqrt{2^n}}\begin{bmatrix}-1\\-1\\-1\\\vdots\\-1\end{bmatrix}\longrightarrow\begin{bmatrix}1\\0\\0\\\vdots\\0\end{bmatrix}$$

均等な関数

$$H^{\otimes n}\frac{1}{\sqrt{2^n}}\begin{bmatrix}-1\\1\\1\\\vdots\\-1\end{bmatrix}\longrightarrow$$

均等のときは
100%の確率で
$|000\ldots0\rangle$ 以外を
計測

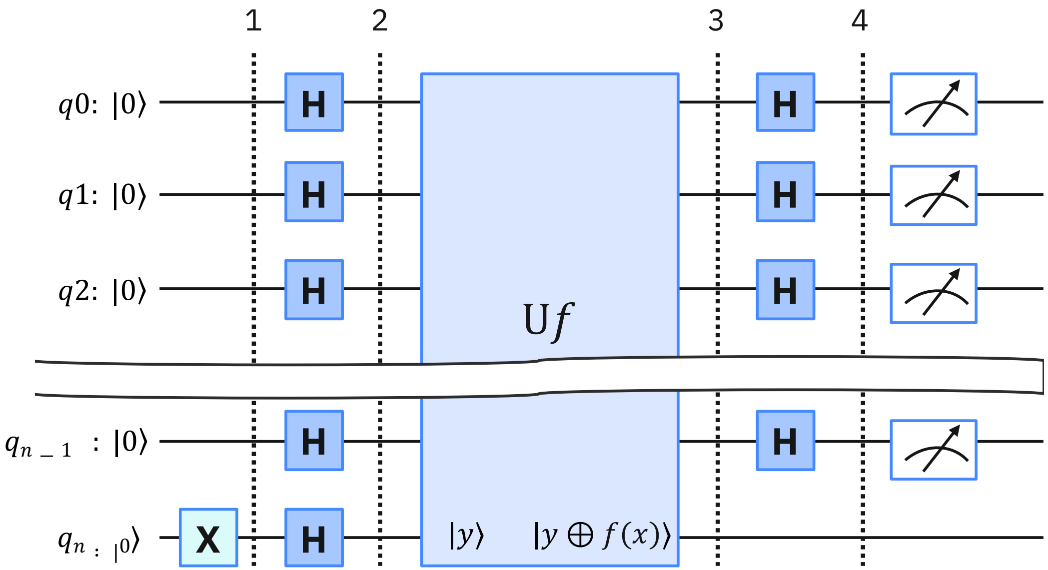
ドイチ・ジョザのQiskitでの実装

1. U_fの実装から考えます

一定な関数

均等な関数

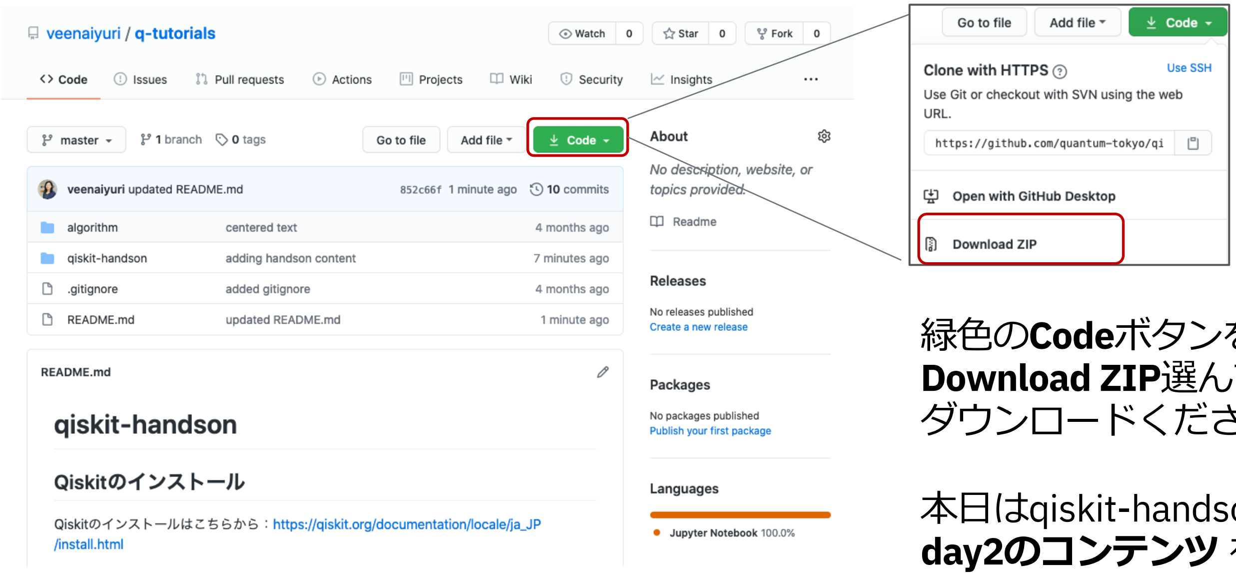
2. アルゴリズム全体の構築
3. シミュレーターでの実行
4. 実機での実行



Qiskitで実装してみよう

Qiskitハンズオン演習にうつります

- 本日のハンズオンは、予め用意されたJupyter Notebookを利用します。
  - https://bit.ly/qiskitho **本日はday2のコンテンツを使います。**



緑色の**Code**ボタンを押して、一番下の**Download ZIP**選んでコンテンツをダウンロードください。

本日はqiskit-handsonというフォルダの中の**day2のコンテンツ** を使います。

実行環境はローカルPCか、IBM Quantum Experienceのどちらかを利用できます。

2.ベルンシュタイン・ヴァジラニアルゴリズム

ベルンシュタイン・ヴァジラニ問題

$$f(x_0, x_1, x_2, \dots) \rightarrow \{0,1\} \quad x_\mathrm{n} \in \{0, 1\}$$

ベルンシュタイン・ヴァジラニ(BV)問題は、ドイチ・ジョザ(DJ)の限定版です。DJが関数の性質を識別する問題であったのに対し、BVは関数の性質がわかっているなかで 関数内に埋め込まれた**秘密の文字列 s を求める問題**です。

関数$f(x)$はs と xのビット毎の2を法とする内積を必ず返すことがわかっています。

$$f(x) = s \cdot x \ (mod\ 2)$$

古典：順番に $|0 \dots 001\rangle$, $|0 \dots 010\rangle$, $|0 \dots 100\rangle$ を問合せて、 s の各ビットを明らかにしていく必要があります。nビットのときの 問合せ量はn回になります。

$$f(0 \dots 001) = s_1, \cdots f(1 \dots 000) = s_\mathrm{n}$$

量子：問合せ量は1回で済みます。

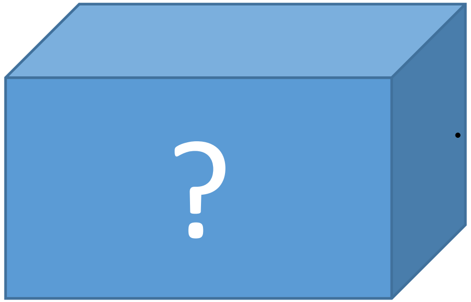
秘密の数

箱の中に秘密の数が隠されているとします。

秘密の数はバイナリ文字列　(i.e. 0100110...)であることだけわかっています。

箱に1回ずつ問い合わせて正しい数字かどうかを答えてもらうことができます。

できるだけ少ない問合せ量で秘密の数字をあてるにはどうしたらよいか？



古典コンピューターの解き方

	$\boxed{011001}$
AND	000001
	000001
AND	000010
AND	000100

⋮

~~000000~~ ?

~~000001~~ ?

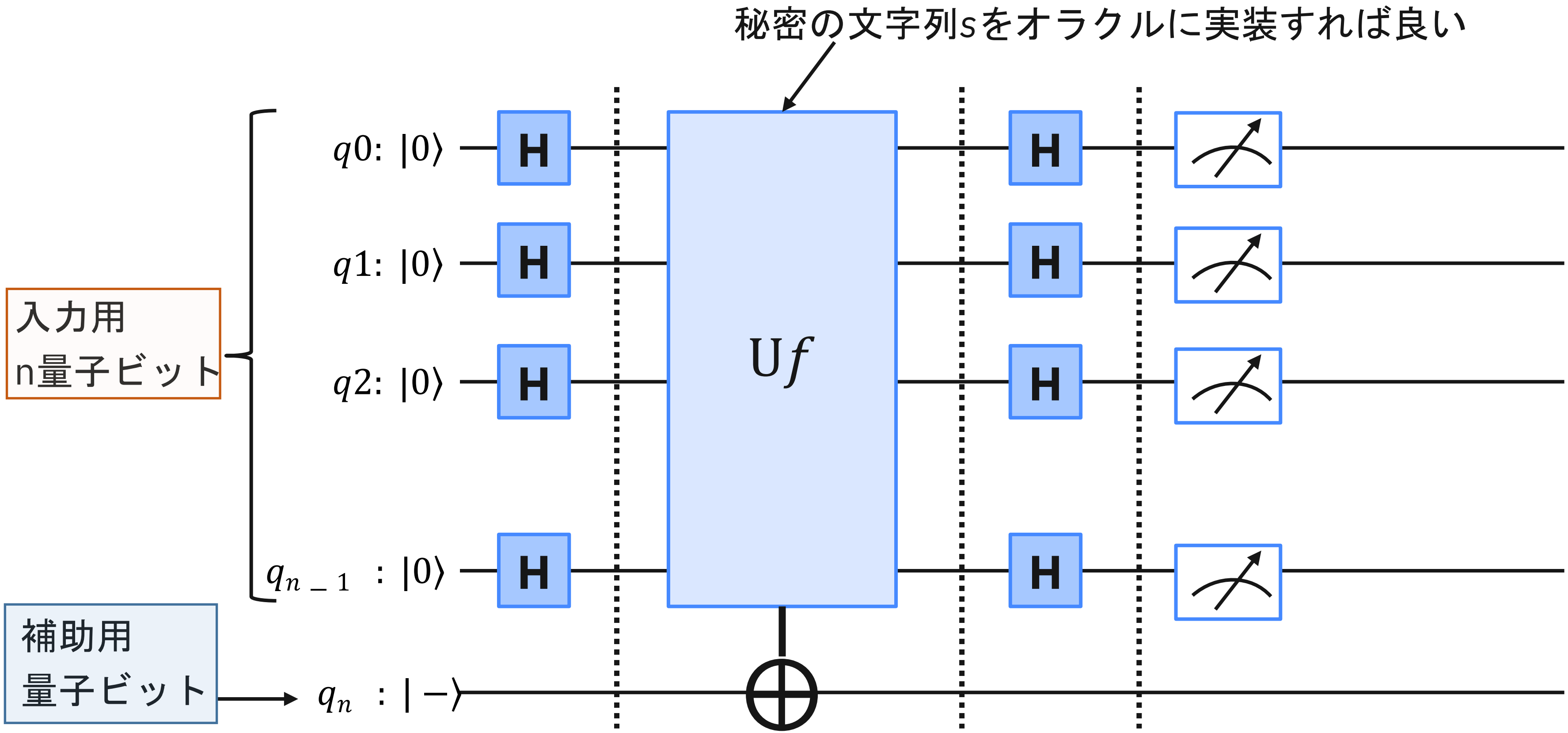
⋮

111111 ?

問合せ量は6回　nビットの文字列だとn回!

ベルンシュタイン・ヴァジラニのアルゴリズム

IBM **Quantum**



ベルンシュタイン・ヴァジラニのアルゴリズム

Step1: 入力用n量子ビット$|0\rangle^{\otimes n}$,アンシラは$|1\rangle$に初期化

Step2: 重ね合わせをつくる　$H^{\otimes n}|0\rangle^{\otimes n} \otimes H|1\rangle = |x\rangle \otimes |-\rangle$

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle|-\rangle$$

量子状態 $|a\rangle$ にHゲートをかける

$H^{\otimes n}|a\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{a\cdot x}|x\rangle$

Step3: オラクルU_fを適用

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}(-1)^{s\cdot x}|x\rangle|-\rangle$$

$s\cdot x$ の各ビットのXORをとった結果から、
n番目のビットの数字が1か0かを判定する.

Step4: n量子ビット全体に再びアダマールをかける

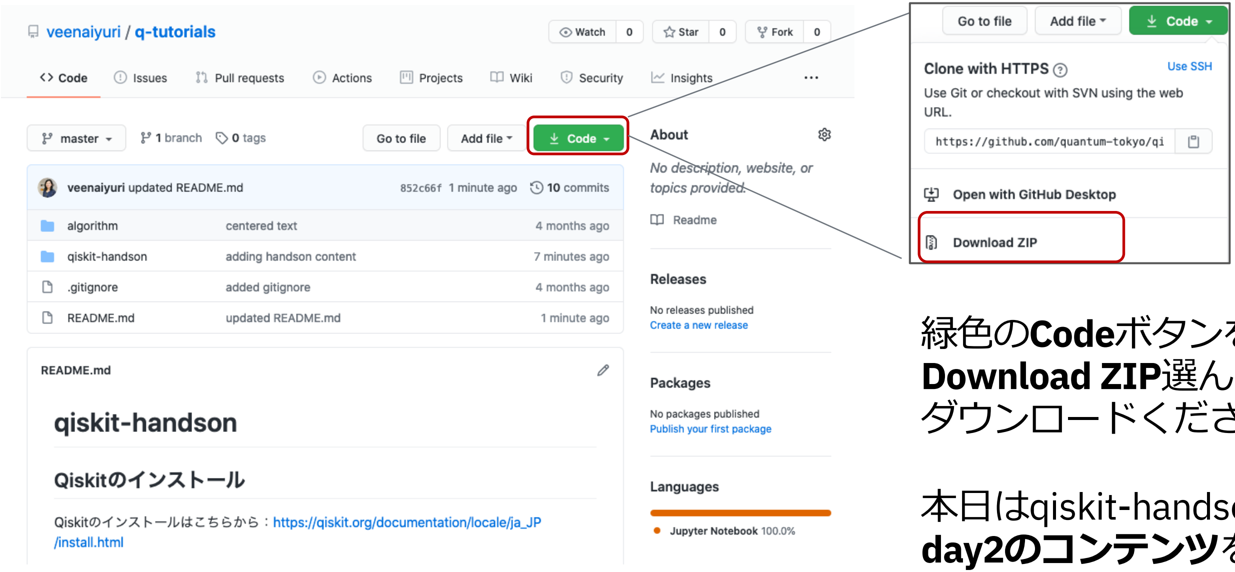
$$|s_1\rangle\cdots|s_n\rangle|1\rangle$$

Step5: n量子ビットだけを測定する

Qiskitで実装してみよう

Qiskitハンズオン演習にうつります

- 本日のハンズオンは、予め用意されたJupyter Notebookを利用します。
  - **https://bit.ly/qiskitho 本日はday2のコンテンツを使います。**



緑色の**Code**ボタンを押して、一番下の**Download ZIP**選んでコンテンツをダウンロードください。

本日はqiskit-handsonというフォルダの中の**day2のコンテンツ**を使います。

実行環境はローカルPCか、IBM Quantum Experienceのどちらかを利用できます。

まとめ

同じ問題を古典アルゴリズムに比べて少ない問合せ量で解くことのできる量子アルゴリズムは量子アドバンテージのあるアルゴリズムと言える。

位相キックバックはオラクルは条件を満たすものを識別する分類機のような役割を果たしてくれる。

位相キックバックは、サイモン、ショア、グローバー、位相推定など多くの有名な量子アルゴリズムで用いられているテクニック。

ドイチ・ジョザとバーンスタイン・ヴァジラニアルゴリズムでは、これらのテクニックを用いることで、古典計算よりも少ない問合せ量で問題を解くことができる。

Thank you

Yuri Kobayashi

yurik@jp.ibm.com

© Copyright IBM Corporation 2020. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. Any statement of direction represents IBM's current intent, is subject to change or withdrawal, and represent only goals and objectives. IBM, the IBM logo, and ibm.com are trademarks of IBM Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available at Copyright and trademark information.