XOR Oracle

考虑布尔函数 $f: \{0,1\}^n \to \{0,1\}^m$，其中 n 是输入比特数，m 是输出比特数。
函数 f 的定义如下：

$$f(x) = \begin{cases} 0, & \text{如果 } x = 0^n \\ 1, & \text{否则} \end{cases}$$

- 优点：
  1. 输入和输出具有相同的比特数 $(n+m)$
  2. 通过构造是可逆的
- 可逆性证明：参见教材第 22 章问题 Q22.2

Phase Oracle

考虑布尔函数 $f: \{0,1\}^n \to \{0,1\}$，其中 n 是输入比特数。
相位量子 Oracle U_f 定义如下：

$$U_f x = (-1)^{f(x)} x$$

其中 $x \in \{0,1\}^n$，x 表示输入量子比特。
相位量子 Oracle 的特点：

- 如果 $f(x)$ 是偶数，则输出与输入相同。
- 如果 $f(x)$ 是奇数，则输出与输入相差一个相位旋转 $e^{i\pi} = -1$。

Phase Oracle - 示例

考虑一个简单的相位量子 Oracle，其中 $f(x) = x \cdot x$。

$$U_f x = (-1)^{x \cdot x} x$$

- 当 $x = 00$ 时，$x \cdot x = 0$，输出为 00。
- 当 $x = 01$ 时，$x \cdot x = 0$，输出为 01。
- 当 $x = 10$ 时，$x \cdot x = 0$，输出为 10。
- 当 $x = 11$ 时，$x \cdot x = 1$，输出为 $-1 \cdot 11 = -11$。

量子算法 - Deutsch 问题

布尔函数：$f: \{0,1\} \to \{0,1\}$
判断一个函数是平衡函数还是恒定函数?
如果对于所有输入，函数满足 $f(0) = 0$ 和 $f(1) = 0$，则该函数为恒定函数。
如果对于所有输入，函数满足 $f(0) = 0$ 和 $f(1) = 1$，或者 $f(0) = 1$ 和 $f(1) = 0$，则该函数为平衡函数。
如果对于所有输入，函数满足 $f(0) = 1$ 和 $f(1) = 1$，则该函数为恒定函数。
经典计算需要进行两次计算来判断一个函数是平衡函数还是恒定函数，而量子计算可以在一次计算中完成。这展示了量子计算的优势和并行性。

Grover's Search Algorithm - 格罗弗搜索算法

- N 个数据在未排序的数据库中
- 索引

$$0, 1, 2, \ldots, N-1$$

- $f(x) = \begin{cases} 1, & \text{if } x = a \\ 0, & \text{if } x \neq a \end{cases}$
- 经典搜索的平均时间复杂度：$O(N)$

Grover's Search Algorithm - 格罗弗搜索算法

- 量子计算中的 Grover's 算法
- 时间复杂度：$O(\sqrt{N})$
- Grover's 算法的工作原理：
  1. 初始化：将数据库中的所有数据通过量子态表示，并放入叠加态中。
  2. 反转：对目标元素应用反转操作，使得其幅值变为负数。
  3. 平均：对所有元素应用平均操作，增加目标元素的幅值。
  4. 重复反转和平均步骤，直到目标元素的幅值接近 1。
  5. 量子测量：对数据库进行测量，得到目标元素的索引。

Grover's Search Algorithm

N 个数据在无序数据库中索引

$$\begin{matrix} 0 \\ 1 \\ 4 \\ 5 \\ \vdots \\ N-1 \end{matrix}$$

$$f(x) = \begin{cases} 1, & \text{如果} x = a \\ 0, & \text{其他情况} \end{cases}$$ 平均时间复杂度: $O(\sqrt{N})$

Grover's Algorithm: Oracle 示例

我们将目标元素 $a = 01001$ 编码为基向量。Oracle 将标记目标元素的量子态，即对应 $f(a) = 1$ 的情况下，对目标元素的量子态进行反转操作。Oracle 的作用：

$$\text{Oracle}xy = \begin{cases} -xy, & \text{如果} x = a \text{且} y = 0 \\ xy, & \text{其他情况} \end{cases}$$

其中，x 表示数据的量子态，y 表示辅助量子态。在这个例子中，Oracle 会在目标元素 a 对应的量子态上施加一个相位反转。

Grover's Algorithm 概述

- 基向量编码：用 (x) 表示数据的量子态，并且目标元素编码为 $|a\rangle$。
- 三个重要的向量：
  - $|a\rangle$：目标元素的量子态。
  - $|\psi\rangle$：与目标垂直的向量。
  - $|f(x)\rangle$：用于标记目标元素的量子态，其中 $f(a) = 1$，而 $f(x) = 0$，如果 $x \neq a$。
- 这三个向量满足以下关系：
  - 正交性：$\langle a|\psi\rangle = 0$。
  - 完备性：$|\psi\rangle + |f(x)\rangle = |x\rangle$。

在 Grover's Algorithm 中，我们的目标是找到目标元素 a 对应的量子态 $|a\rangle$。通过重复应用 Grover 迭代操作，我们可以增加目标元素的振幅，并逐渐增加其概率。

Grover 迭代操作

Grover 迭代操作的表达式为：

$$G = -HSO$$

其中，H 表示 Hadamard 变换，S 表示相位反转操作，O 表示 Oracle 操作。重复应用 Grover 迭代操作约 $O(\sqrt{N})$ 次后，我们有很高的概率测量到目标元素 a 对应的量子态。这使得 Grover's Algorithm 的时间复杂度约为 $O(\sqrt{N})$，相较于经典搜索算法的 $O(N)$ 来说，有明显的加速效果。

Grover's Algorithm 概述 (续)

- 三个重要的向量：
  - $|\psi\rangle$：等概率叠加向量，即所有数据的量子态的等概率叠加。
  - $|a\rangle$：目标元素的量子态。
  - $|\psi_0\rangle$：和 $|a\rangle$ 垂直的向量。由于 $|\psi\rangle$ 是其他向量的线性组合，所以这三个向量都在同一个平面上。
- 这三个向量的关系可以表示为：
  - 正交性：$\langle a|\psi\rangle = 0$。
  - 完备性：$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_x |x\rangle$。
  - 补集关系：$|\psi_0\rangle = \frac{1}{\sqrt{N}} \sum_{x \neq a} |x\rangle$。

Grover 迭代操作 (续)

Grover 迭代操作涉及两个算子 V 和 W:

- 应用算子 V: 将 $|\psi\rangle$ 绕 $|a\rangle$ 的镜像翻转。
- 应用算子 W: 将新的向量绕 $|\psi_0\rangle$ 的镜像翻转，使其更接近目标元素 $|a\rangle$。

重复应用 Grover 迭代操作，目标元素 $|a\rangle$ 的概率会逐渐增加，从而最终能以高概率找到目标元素。

在 V 中嵌入 $f(x)$

在 Grover's Algorithm 中，我们需要一个 Oracle 操作来实现函数 $f(x)$ 的嵌入。Oracle 操作 O_f 对量子态 $|x\rangle$ 的作用是：

$$O_f|x\rangle = (-1)^{f(x)}|x\rangle$$

因此，当 $f(x) = 1$ 时，Oracle 会给 $|x\rangle$ 的相位加上 -1。这等效于在 $|a\rangle$ 的反射，从而将 $|a\rangle$ 嵌入到了 V 中。

Grover's Algorithm - V 算子实现 (续)

在 Grover's Algorithm 中，V 算子的实现如下：

$$V = I - 2|\psi\rangle\langle\psi|$$

其中，I 是单位矩阵，$|\psi\rangle$ 是等概率叠加向量。
例如，对于 2 个数据的情况，$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |01\rangle)$，则 V 算子的实现为：

$$V = I - 2|\psi\rangle\langle\psi| = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

V 算子保持所有基向量不变，但将目标向量 $|a\rangle$ 的符号取反。