The lecture went on to trace state diagrams like the qube from the text ch. 8, showing how cancellation only to interference can be targeted to pile up the non-zero amplitude on to other nodes designated for acceptance. Depending on $U_2$ & $U_1$ vs. $U_3d$ & $U_{1d}$, the amplitude piles up on the upper two nodes (qubit $i = 0$) or the lower two nodes. Getting such a “switcheroo” to reflect desired accept/reject criteria is the essence of designing quantum algorithms for decision problems. For functions $f(x) = y$, the goal is to pile up $(1-\varepsilon)$ amplitude on the outcome $y$, but what more often happens is sampling from a distribution in which multiple outputs are possible.

Ch. 9 extends Deutsch’s task from 1+1 qubits to $n+1$ qubits. Now it takes $2^n + 1$ evaluations to conclusively tell whether $f(x_1, -x_0)$ is constant versus it being balanced (presuming it is one or the other) in a classical setting. But, in a quantum setting starting with $|\psi\rangle \otimes |0^{n+1}\rangle$, applied to $|0^{n+1}\rangle$. The final $\langle f \rangle$ picks up a minus sign that spurs cancellations. The effects are similar:

- The “quantum advantage” is exponential in $n$ (though the criterion is contrived).
- “Maple diagrams” don’t scale, but the linear algebra calculations remain tractable.
That was as far as I got (orally) in the lecture. The intended endpoint was Ch 10: Simon's Algorithm in which the two situations being distinguished are \( f \) is \( 1 \rightarrow 1 \) vs. \( f \) is \( 2 \rightarrow 1 \):

There is a "hidden vector" \( S \in \{0,1\}^n \) such that for all vectors \( \gamma, \delta \in \{0,1\}^n \):

\[
f(\gamma) = f(\delta) \iff \gamma \oplus S = \delta \oplus S,
\]

bitwise XOR

If \( S = 0^n \) then the RHS is \( \gamma = \delta \) so it says \( f(\gamma) = f(\delta) \iff \gamma = \delta \) so \( f \) permutes \( \{0,1^n\} \).

Else \( S \) defines a "cleft" in \( \{0,1^n\} \) in the following sense: \( \{0,1^n\} = \mathbb{A} \cup \mathbb{B} \) such that \( \mathbb{B} = \{v \oplus S: v \in \mathbb{A}\} = \text{det } \mathbb{A} + S \) and \( f \) behaves identically (and injectively) on \( \mathbb{A} \) vis-à-vis \( \mathbb{B} \). How quickly can we tell whether a given \( f \) has such a cleft?

**Theorem:**

- A classical randomized algorithm — with \( f \) given only as a "black box" to get values \( f(\gamma) \) given binary — needs exponential time to tell with high probability.

- Whereas a quantum algorithm can compute in equations defining \( S \) in expected time \( O(n^2) \) iterations \( \times O(n) \) work per iteration. [O. Simon 1992]

This does not imply \( \text{BQP} \neq \text{BPP} \) because of the black-box condition on how \( f \) is accessed — which even allows \( f \) to be non-computable! But it is the main proven result of that character. And it inspired Peter Shor to replace the initial Hadamard transform by the n-qubit Quantum Fourier Transform to see what happens — leading to Shor's quantum factoring algorithm (1993-94). Taken together, can this "Quantum Advantage for Hidden Subgroups" phenomenon be understood further?