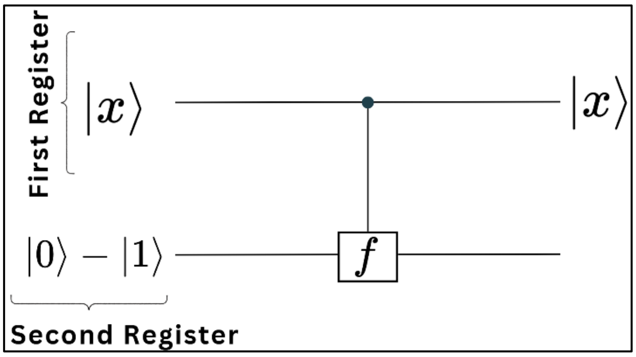
# The supremacy of quantum algorithms
# Lesson 2.1–Quantum Function Evaluation

The core of quantum algorithms lies in the use of interference, which is one of the fundamental properties of quantum mechanics. This interference is achieved by applying a function $f$ to the entries of the first register. In this lesson, you will learn exactly how this process works.

In the first register, you have the state $|x\rangle$, where $x$ is either 0 or 1. The function $f$ acts as follows

- If $f(x) = 0$, the states of the second register remain unchanged.
- If the function $f(x) = 1$, a state $|0\rangle$ in the second register transforms into $|1\rangle$, and a state $|1\rangle$ into $|0\rangle$.




The fact that the second register can be transformed according to the value of the function $f$ applied to the first register is represented schematically in the picture above. This is described as the first register controlling the second one.[1]

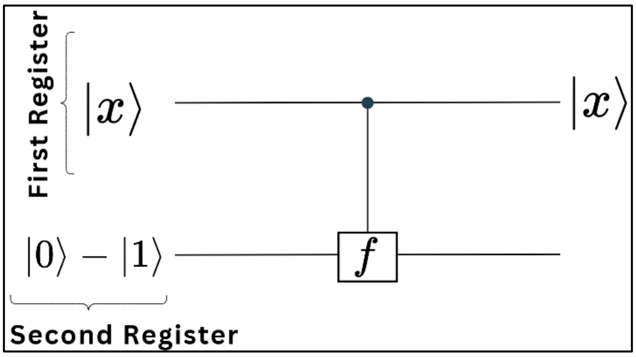
**Task 1:** Apply the method described above, where the function $f$ acts on the first register and potentially modifies the values of the second register. Complete the following table by determining the values of the second register based on the given function values.

| Function value applied to the first register $f(x)$ | Initial state of the second register | Final state of the second register (controlled by $f$) |
|---|---|---|
| 0 | $|0\rangle$ | |
| 0 | $|1\rangle$ | |
| 1 | $|0\rangle$ | |
| 1 | $|1\rangle$ | |
| 0 | $|0\rangle - |1\rangle$ | |
| 1 | $|0\rangle - |1\rangle$ | |

[1] Mathematically, this operation corresponds to a so-called *controlled modulo-2 addition* (XOR operation), which can be expressed as follows: $|x\rangle|y\rangle \longmapsto |x\rangle|y \oplus f(x)\rangle$

Here, $\oplus$ denotes addition modulo 2. This means: If $f(x) = 1$, the state of the second qubit $|y\rangle$ is flipped ($|0\rangle \leftrightarrow |1\rangle$); if $f(x) = 0$, the state remains unchanged.

**Task 2:** In this task, you will use the results from Task 1 to understand the key trick used in quantum algorithms to introduce interference, which is the core of quantum computing. Quantum algorithms use the following circuit for a function $f$ that acts on the first register while controlling the second register.




a) Show that if $f(x) = 0$, then the full state

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

remains the same.

b) Show that if $f(x) = 1$ , then the full state transforms $|x\rangle\,(|0\rangle - |1\rangle)$ transforms into

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

and that this can be written as:

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

c) Justify that the results from **a)** and **b)** are contained in the following general expression:

$$(-\mathbf{1})^{f(x)}|x\rangle\,(|1\rangle - |0\rangle)$$

**Conclusion:** This means that the second register remains completely unchanged, but the first register gains a phase factor of $(-1)^{f(x)}$.

**Final Conclusion: The Role of Quantum Function Evaluation in Quantum Algorithms**

In this lesson, you learned how quantum function evaluation introduces interference, the core mechanism behind quantum algorithms. Instead of simply modifying a qubit's state, the function encodes information in the phase of the first register.

The state $|0\rangle - |1\rangle$ in the second register is crucial for achieving this interference. It ensures that function evaluation does not store information classically but instead modifies the phase of the first register.

Since this phase encoding is present in most quantum algorithms, understanding it is essential for both the mathematical formulation of quantum computing and its practical implementation in tools like Quantum Composer or Qiskit. Mastering this principle will help you understand how quantum circuits solve problems more efficiently than classical algorithms.

In the next lessons, you will see how this technique is applied in real quantum algorithms!