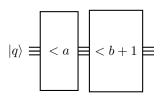
Documentation of Implementation A

Documentation for implementing the oracle 'Interval [a, b]' using the implementation A (previously described).

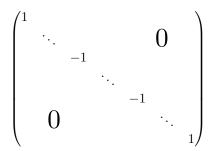
Oracle as Black Box



Oracle as its components



Unitary Matrix of Oracle



Ideas inspiring the oracle

This oracle reuses the less-than oracle. It applies them twice to give a π -phase to the desired states and give a 2π -phase (return to initial state) to other ones. Firstly it applies the oracle 'less-than a', and then the oracle 'less-than b+1', marking all the states in the interval [a,b].

Classical algorithm which builds the oracle

Parameters needed for the classical algorithm which builds the oracle.

Parameters of the function:

- a: Lower boundary of the range of integers.
- b: Upper boundary of the range of integers.
- n: Number of qubits.

Parameters of the oracle:

 Which qubits of the general circuit is the oracle applied to.

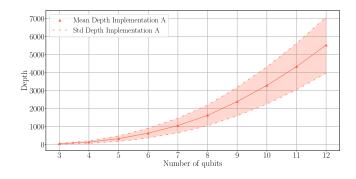
Conditions:

- Precondition: There are no hard preconditions on the states.
- Postcondition: Input state with a π -phase applied to states within range [a, b].

Oracle Circuit

The details in this section are given with respect to a specific backend. In this case, FakeWashingtonV2 from Qiskit.

Depth:



Gate Set:

- The oracle requires a universal gate set (Clifford and T gates).
- The backend FackeWashingtonV2 has the following gate set: CX, RZ, S, X.

Assumptions over connections:

• The oracle assumes that each qubit is connected to the rest of the qubits.