- Middle East Technical University   Department of Computer Engineering

CENG 487

Introduction to Quantum Computing

Fall 2021-2022

Assignment 2

Due date: 25.12.2021, 23:59

1. The Environment

a. You will implement quantum computing programs using real quantum computers on IBM quantum computing environment(quantum-computing.ibm.com).

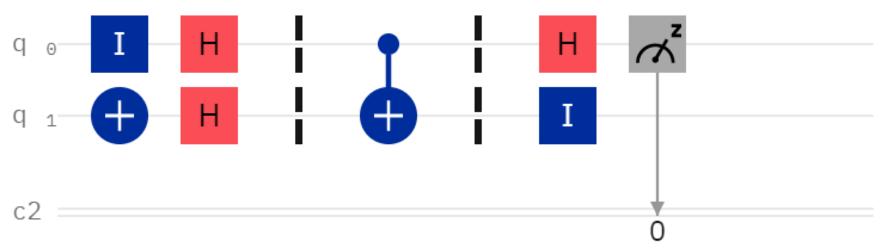
b. The easiest way to create your quantum circuits visually or using OpenQASM 2.0 (an assemly-like language for quantum computing) is IBM Quantum Composer(quantum-computing.ibm.com/composer). You can also use python library Qiskit (v0.3.2.0) on IBM Quantum Lab(https://lab.quantum-computing.ibm.com)

c. You can get help from the tutorials to start with quantum computing: quantum-computing.ibm.com/docs/

2. Examples on Quantum Composer

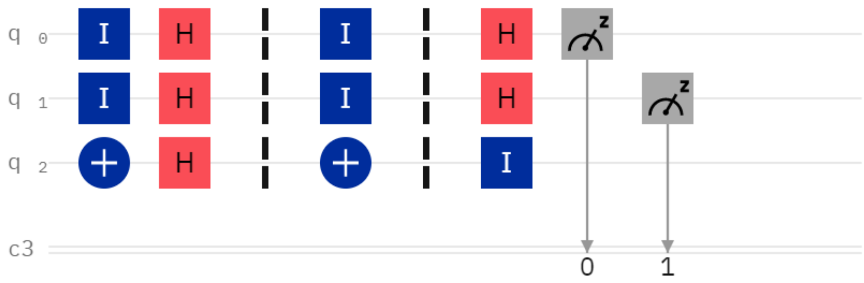
Recall Deutsch's Oracle and Deutsch-Jozsa Algorithm. In order to check for a function $f:\{0,1\}^n \to \{0,1\}$ to be constant or balanced in 1 query, we can construct an oracle circuit U_f such that it outputs $x \in \{0,1\}^n$ for the first n qubits and $y \oplus f(x)$ for the last qubit while the inputs are $x \in \{0,1\}^n$ and $y \in \{0,1\}$

Example a: $f(x) = x, n = 1$



In Example a, q_1 is used as the control bit, starting from $|1\rangle$. U_f is the circuit between barriers. CNOT gate is applied inside U_f so that the output will become $y \oplus f(x) = y \oplus x$. By measuring q_0 (which is equal to x) we can tell whether $f:\{0,1\} \to \{0,1\}$ is constant or balanced. Check course slides for further details.(Identity matrices are just to show the operations applied on the system as a whole and visual purposes. If you try without Identity matrices and the barriers the result will not be changed)

Example b: $f(x) = 1, n = 2$



In Example b, q_2 is used as the control bit, starting from $|1\rangle$. U_f is the circuit between barriers. NOT gate is applied inside U_f so that the output will become $y \oplus 1 = \neg y$. By measuring q_1 and q_0 ($x = q_1 q_0$) we can tell whether $f: \{0,1\}^2 \rightarrow \{0,1\}$ is constant or balanced. Check course slides for further details.(Identity matrices are just to show the operations applied on the system as a whole and visual purposes. If you try without Identity matrices and the barriers the result will not be changed)

3. Deutsch-Jozsa Algorithm

At each part, you will construct a quantum circuit to check for a function $f: \{0,1\}^n \rightarrow \{0,1\}$ to be constant or balanced (2 in total), and test your circuit. (for every x, x_0 is the least significant bit whereas x_{n-1} is the most significant one. You can choose qubits such that q_i represents x_i, where $i \in \{0,1,\dots,n-1\}$)

Note: It is not mandatory but you may want to run the circuit also on a simulator to check whether some possible unexpected outcome are caused by the wrong implementation or it is just the system. You can include this checks on your report. But make sure that you include your work about the results on real systems.

a. $f(x) = x_0 \oplus x_1 \oplus x_2, n = 3$

Implement the quantum circuit to check whether the function above is constant or balanced, and run it on a system with shots=1024.
c.1. What are the results?
c.2. Is the function constant or balanced? Clearly explain how you derive your answer?

b. $f(x) = x_0 \oplus x_3 \oplus (x_1 \cdot x_2), n = 4$

Implement the quantum circuit to check whether the function above is constant or balanced, and run it on a system with shots=1024.
c.1. What are the results?
c.2. Is the function constant or balanced? Clearly explain how you derive your answer?

4. Submission

a. For this assignment, we **expect** you to upload your report as a **pdf** file consisting of your answers, comments and comparisons; the **images** and OpenQASM or Qiskit **codes** of your circuits.

b. Try to finish as early as possible. The sytems to run your work are shared between many different users across the earth. Therefore some systems may have longer queues that can cause for you to wait. (look for systems that have smaller number of jobs on queue).

c. You can still submit your work if the **deadline** is passed, however with an increasing **penalty** of **5*days*days**. (i.e. first day -5 points, second day -5*2*2=-20 points and so on). Note that even a minute late means that it is the other day.

d. The assignments are for individual work. You work should be done by only you and it should be genuine. We have zero tolerance policy for cheating. People involved in cheating will be punished according to the university regulations and will get zero.