Student Information

Name: Kaan Karaçanta

ID: 2448546

Part 1

a)

For this function, $c = 0^n$ would make it one-to-one.

b)

The minimum number of evaluations required to determine exactly whether the function is two-to-one or one-to-one is $2^{n-1} + 1$, so 65.

c)

This is a two-to-one function, 000 and 011 are one pair of inputs giving the same result, their xor is 011, so $c = 011$. This relationship can be seen on any other pair of inputs giving the same result.

d)

Without knowledge of which specific inputs correspond to the output measurements, we cannot directly determine the secret string c. Simon's algorithm relies on finding n linearly independent equations that result from measuring the quantum state after applying the quantum oracle and subsequent Hadamard gates. These equations are derived from the bitstrings that are guaranteed to be orthogonal to the secret string c.

In this case, while we have 8 measurements with 7 different results, which should theoretically be more than enough for an n=6 bit string, they do not form a set of linearly independent equations. The lack of linear independence means that the measurements do not span the full vector space and therefore do not provide a complete basis to solve for c.

*This answer is wrong it can be found.

e)

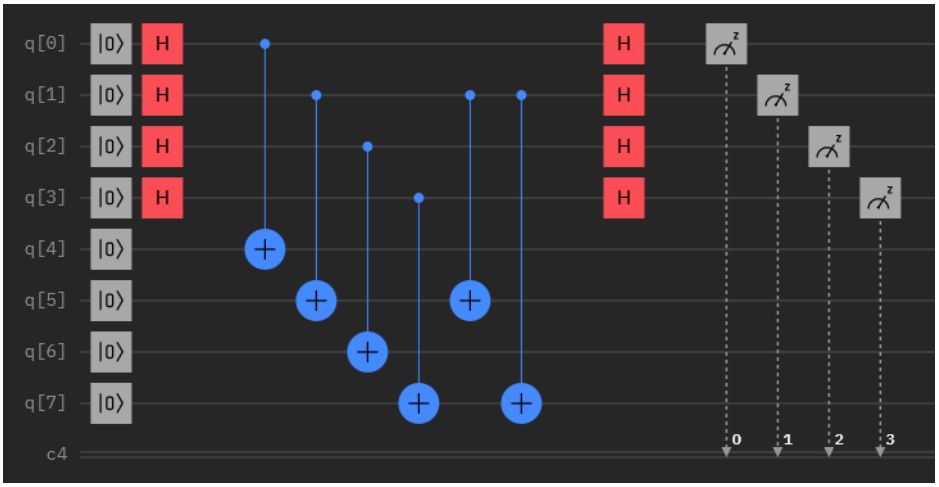

Figure 1: Simon's Algorithm Circuit with $c = 1010$

For Simon's Algorithm with $n = 4$ we need to initialize the first $2n$ qubits to $|0\rangle$ and apply Hadamard gate to the first 4 of them, which can be called as the first register. Then, the oracle is applied by applying CNOT gates between 2 registers to create a superposition of states that encode the function, and then applying CNOT gate between $q[1]$, $q[5]$ and $q[1]$, $q[7]$ for the specific c in our function. After that, we apply Hadamard gate to the first register again and measure the first 4 qubits.

To change the circuit to $c = 0101$, we need to change the CNOT gates between $q[1]$, $q[5]$ and $q[1]$, $q[7]$ to $q[0]$, $q[4]$ and $q[0]$, $q[6]$. Here is how the circuit would look like:

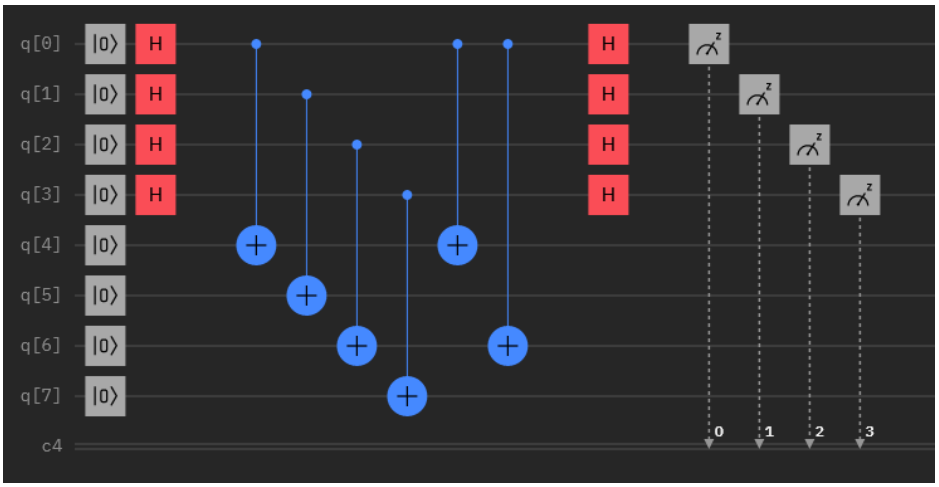

Figure 2: Simon's Algorithm Circuit with $c = 0101$

Part 2

a)

"Majority voting" is an error correction technique used in classical computation where the same data is replicated across multiple, redundant systems. When a read operation occurs, all copies are read, and the value that appears most frequently (the majority) is considered the correct value. This method is effective in correcting single errors in the data, assuming that the majority of the copies are correct, like sending 111 for just a single bit 1 even if it will be recieved as 101, since the majority is still 1, it will be considered as 1, so in this way the chance of error, which is higher for just one bit, is reduced.

In quantum computing, majority voting as it is classically understood cannot be directly applied because of the non-cloning theorem, which states that it is impossible to create an identical copy of an arbitrary unknown quantum state. Furthermore, bit flips are not the only possile errors in quantum computing, like phase-flips.

b)

My id is 2448546, and sum of its digits is 33. The phase gate will be $P(\frac{46\pi}{33})$. The angle θ is 4.379 radians, and the cosine of this angle is -0.327. When these gates are applied respectively, after first H gate, we have $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. After the phase gate, we have $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle)$. Then again after the H gate, we have $|\psi\rangle = \frac{1}{2}(1+e^{i\theta})|0\rangle + \frac{1}{2}(1-e^{i\theta})|1\rangle$. From there, the probability of measuring $|0\rangle$ is 0.336 and the probability of measuring $|1\rangle$ is 0.664. When I run this with the simulator, the results are similar to the expectations, 0.335 and 0.665, as follows:

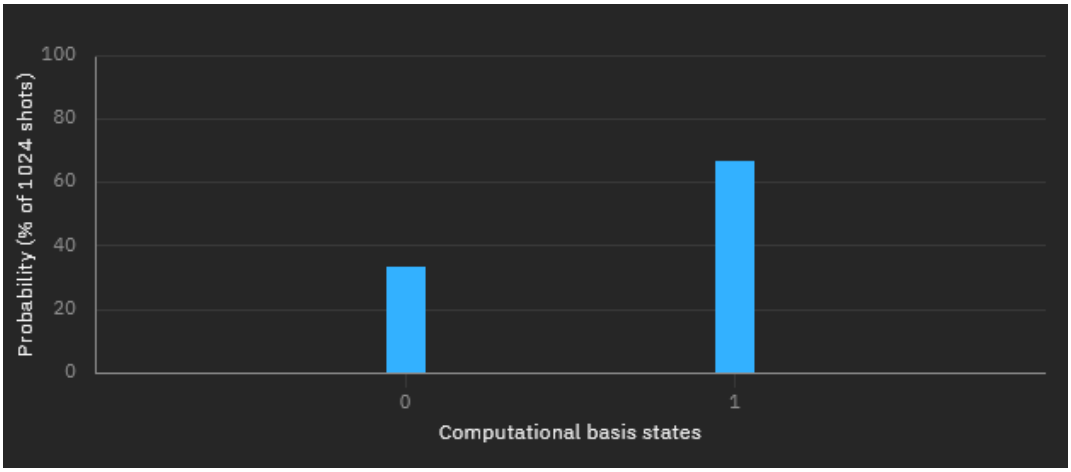

Figure 3: Results of the circuit with the simulator

c)

Here is the first circuit and its result:

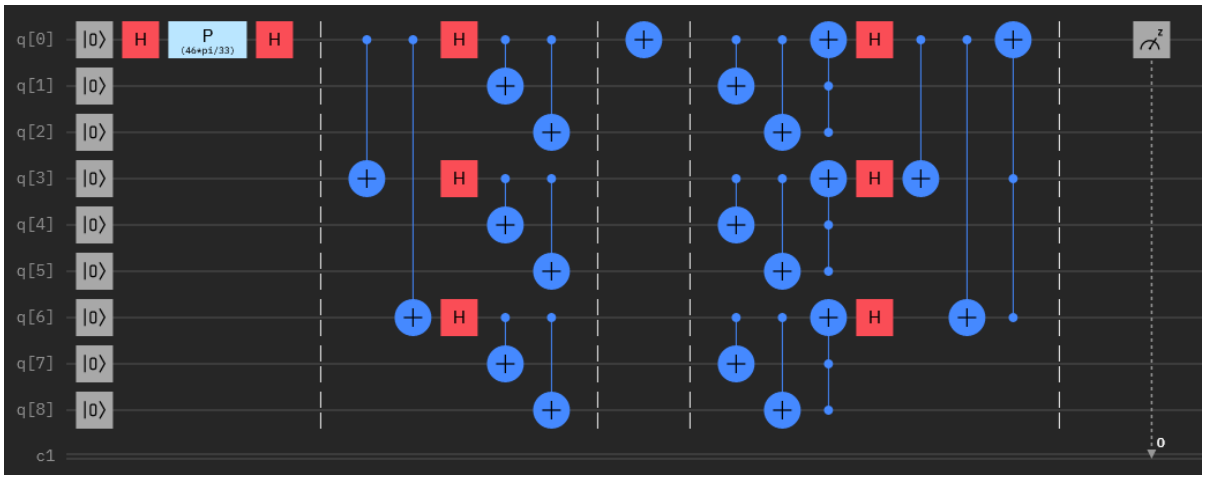

Figure 4: The first circuit

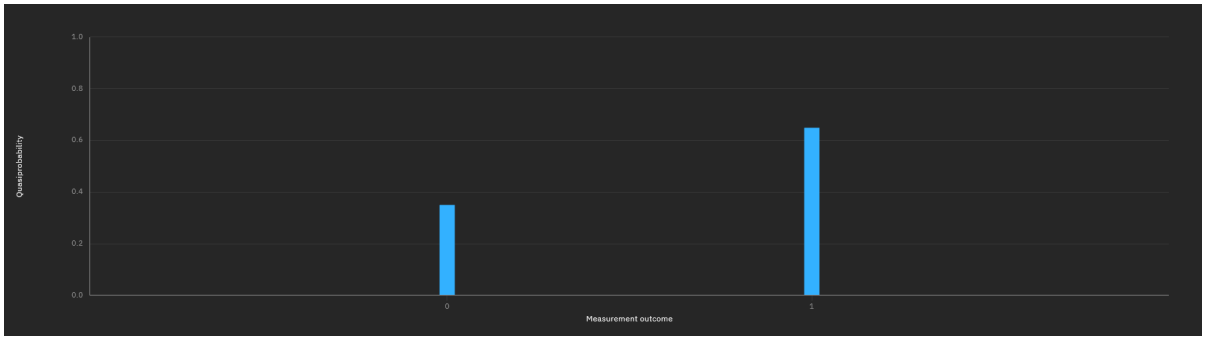

Figure 5: The result of the first circuit

Firstly, I initialized the first qubit with the value I found on the previous part. Then, in the first area separated with barriers, error encoding is implemented, then the error is injected, and finally, error decoding is implemented on the last separated area.

With these steps of error correction, the new results are 0.351 for $|0\rangle$ and 0.649 for $|1\rangle$. The circuit seems correct, yet someshow the results are slightly worse than the one without error correction on the previous part.

Here is the second circuit and its result:

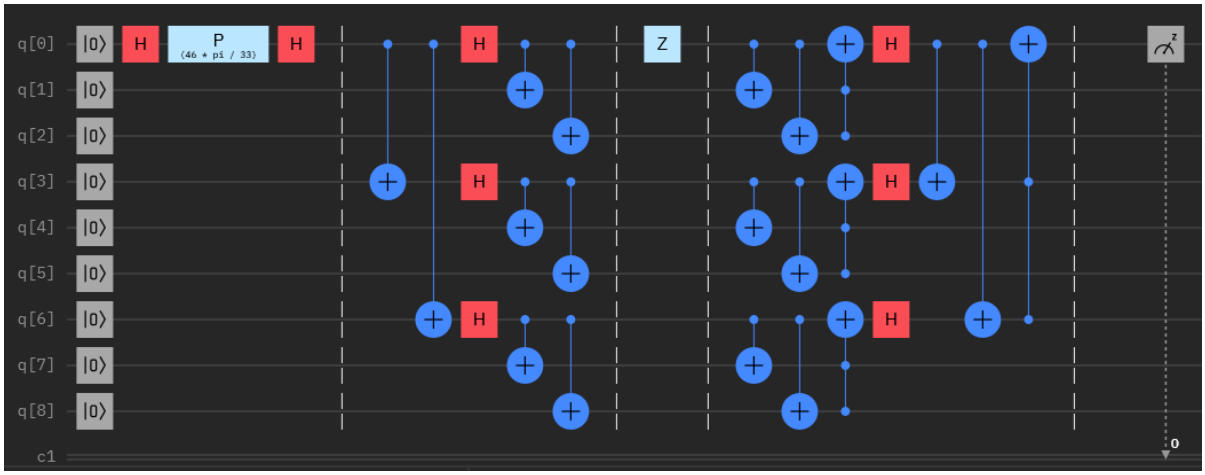

Figure 6: The second circuit

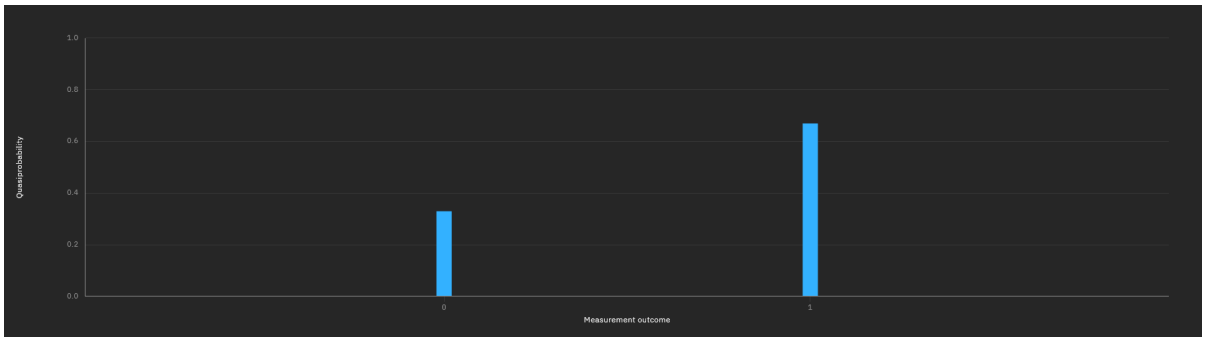

Figure 7: The result of the second circuit

The encoding and decoding are the same as the previous circuit, this time instead of NOT gate there is a Z gate. The results are 0.33 and 0.67, which are slightly better than the previous circuit and closer to the expected results.

Here is the third circuit and its result:

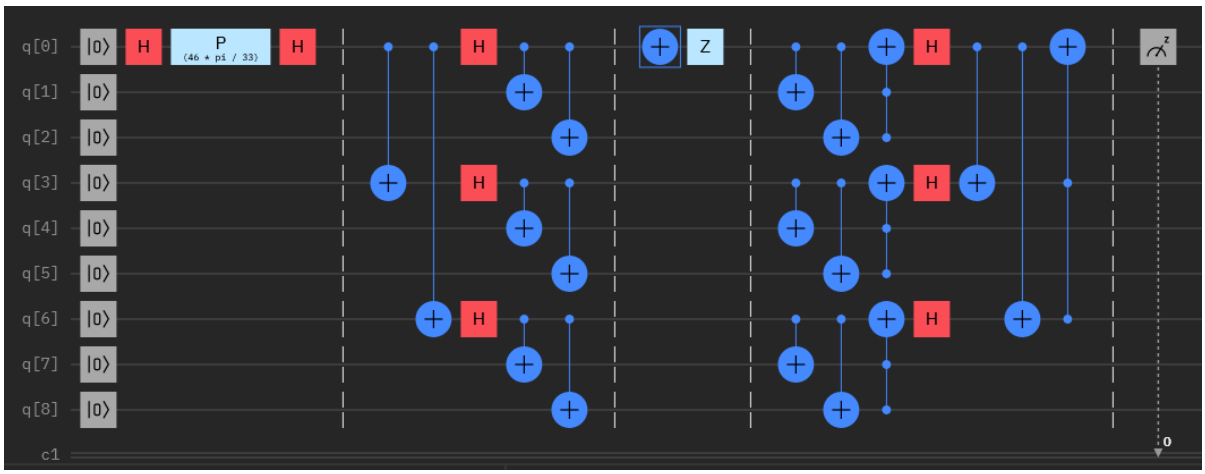

Figure 8: The third circuit

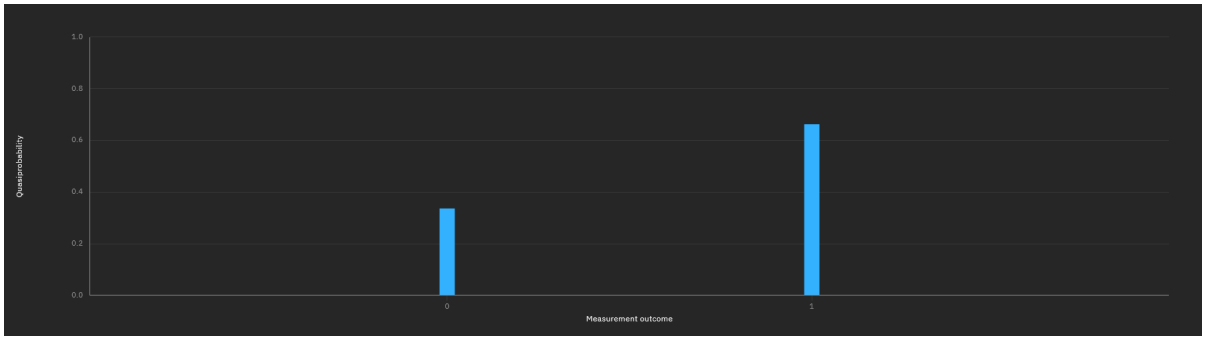

Figure 9: The result of the third circuit

The encoding and decoding are the same as the previous circuit, this time both of the gates are injected in the error area. The results are 0.337 and 0.663, which are the best results among the three circuits, and the closest to the expected results.