Name: **Huan Q. Bui**
Course: **8.370 - QC**
Problem set: **#3**
Due: Wednesday, Oct 5, 2022
Collaborators/References: Christina Yu (alternative approach to 5b, by Nielsen and Chuang)

1. Find a matrix whose square is σ_x.

We can easily find a matrix whose square is σ_z:

$$M = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$$

Now, since $H\sigma_z H = \sigma_x$, where H is the Hadamard, consider the matrix

$$N = HMH = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 1+i & 1-i \\ 1-i & 1+i \end{pmatrix}$$

We find that

$$N^2 = \frac{1}{4}\begin{pmatrix} 1+i & 1-i \\ 1-i & 1+i \end{pmatrix}\begin{pmatrix} 1+i & 1-i \\ 1-i & 1+i \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \sigma_x.$$

So N is a matrix whose square is σ_x.

2. Making a SWAP gate.

Consider two CNOT gates where different qubits are taken as the control qubit:

$$CNOT_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad CNOT_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

The SWAP gate is then given by

$$SWAP = CNOT_2 \cdot CNOT_1 \cdot CNOT_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \checkmark$$

3. Controlled Hadamard gate.

One way to construct this gate is first SWAP 1 and 2, then apply the controlled-Hadamard on the last two qubits (with qubit 3 being the control qubit), then SWAP 1 and 2 again:

$$CH_{c=3,t=1} = SWAP_{12}CH_{c=3,t=2}SWAP_{12} = (SWAP_{12} \otimes \mathcal{I}_3)(\mathcal{I}_1 \otimes CH_{c=3,t=2})(SWAP_{12} \otimes \mathcal{I}_3)$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/\sqrt{2} & 0 & 0 & 0 & 1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\sqrt{2} & 0 & 0 & 0 & 1/\sqrt{2} \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1/\sqrt{2} & 0 & 0 & 0 & -1/\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1/\sqrt{2} & 0 & 0 & 0 & -1/\sqrt{2} \end{pmatrix}$$

Here we have used

$$CH_{c=3,t=2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/\sqrt{2} & 0 & 1/\sqrt{2} \\ 0 & 0 & 1 & 0 \\ 0 & 1/\sqrt{2} & 0 & -1/\sqrt{2} \end{pmatrix}.$$

4. The Fredkin gate.

(a) The Fredkin gate is a controlled SWAP gate. If $x = 0$, we do nothing. If $x = 1$, we swap y and z.

- AND gate: obtained by fixing $y_{\text{in}} = 0$. $(x_{\text{in}}, 0, z_{\text{in}}) \longrightarrow (x_{\text{in}}, \text{output}, \text{junk})$. Truth table:

$$\begin{gathered} (\mathbf{0}, 0, \mathbf{0}) \longrightarrow (0, \mathbf{0}, 0) \\ (\mathbf{0}, 0, \mathbf{1}) \longrightarrow (0, \mathbf{0}, 1) \\ (\mathbf{1}, 0, \mathbf{0}) \longrightarrow (1, \mathbf{0}, 0) \\ (\mathbf{1}, 0, \mathbf{1}) \longrightarrow (1, \mathbf{1}, 0) \end{gathered}$$

- OR gate: obtained by fixing $y_{\text{in}} = 1$. $(x_{\text{in}}, 1, z_{\text{in}}) \longrightarrow (x_{\text{in}}, \text{junk}, \text{output})$. Truth table:

$$\begin{gathered} (\mathbf{0}, 1, \mathbf{0}) \longrightarrow (0, 1, \mathbf{0}) \\ (\mathbf{0}, 1, \mathbf{1}) \longrightarrow (0, 1, \mathbf{1}) \\ (\mathbf{1}, 1, \mathbf{0}) \longrightarrow (1, 0, \mathbf{1}) \\ (\mathbf{1}, 1, \mathbf{1}) \longrightarrow (1, 1, \mathbf{1}) \end{gathered}$$

- NOT gate: Fixing $y_{\text{in}} = 1$ and $z_{\text{in}} = 0$. $(x_{\text{in}}, 1, 0) \longrightarrow (\text{junk}, \text{output}, \text{junk})$. Truth table:

$$\begin{gathered} (\mathbf{0}, 1, 0) \longrightarrow (0, \mathbf{1}, 0) \\ (\mathbf{1}, 1, 0) \longrightarrow (1, \mathbf{0}, 1) \end{gathered}$$

- FANOUT gate: Fixing $y_{\text{in}} = 1$ and $z_{\text{in}} = 0$ just like in the NOT case. Except that we use different bits for our output. $(x_{\text{in}}, 1, 0) \longrightarrow (\text{output}, \text{junk}, \text{output})$. Truth table:

$$\begin{gathered} (\mathbf{0}, 1, 0) \longrightarrow (\mathbf{0}, 1, \mathbf{0}) \\ (\mathbf{1}, 1, 0) \longrightarrow (\mathbf{1}, 0, \mathbf{1}) \end{gathered}$$

(b) We show that the Fredkin gate preserves the number of 1s in the system by exhaustion. Truth table:

$$\begin{gathered} (0, 0, 0) \longrightarrow (0, 0, 0) \\ (0, 0, 1) \longrightarrow (0, 0, 1) \\ (0, 1, 0) \longrightarrow (0, 1, 0) \\ (0, 1, 1) \longrightarrow (0, 1, 1) \\ (1, 0, 0) \longrightarrow (1, 0, 0) \\ (1, 0, 1) \longrightarrow (1, 1, 0) \\ (1, 1, 0) \longrightarrow (1, 0, 1) \\ (1, 1, 1) \longrightarrow (1, 1, 1) \end{gathered}$$

Notice that the number of 1s is preserved in all cases, as desired.

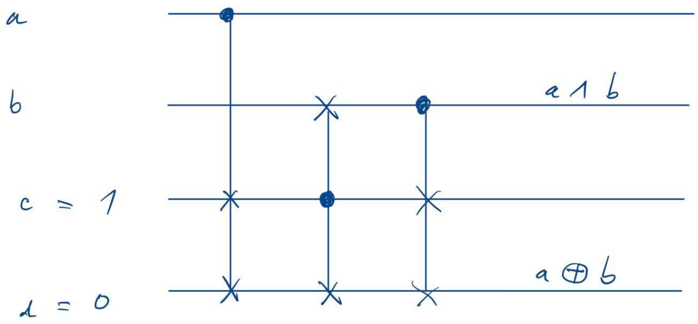
(c) The half-adder is defined by

$$(x, y) \longrightarrow (x \wedge y, x \oplus y)$$

The truth table for the half-adder is

$$\begin{aligned}(0,0) &\longrightarrow (0,0)\\(0,1) &\longrightarrow (0,1)\\(1,0) &\longrightarrow (0,1)\\(1,1) &\longrightarrow (1,0)\end{aligned}$$

Consider the following circuit using 03 Fredkin gates and 4 bits a, b, c, d, where a, b are inputs and $c = 1, d = 0$ are ancillary bits. The output bits will be (b, d).



The circuit works as follows:

$$\begin{aligned}(a,b,c,d) &= (\mathbf{0},\mathbf{0},1,0) \longrightarrow (0,0,1,0) \longrightarrow (0,0,1,0) \longrightarrow (0,\mathbf{0},1,\mathbf{0})\\(a,b,c,d) &= (\mathbf{0},\mathbf{1},1,0) \longrightarrow (0,1,1,0) \longrightarrow (0,0,1,1) \longrightarrow (0,\mathbf{0},1,\mathbf{1})\\(a,b,c,d) &= (\mathbf{1},\mathbf{0},1,0) \longrightarrow (1,0,0,1) \longrightarrow (1,0,0,1) \longrightarrow (1,\mathbf{0},0,\mathbf{1})\\(a,b,c,d) &= (\mathbf{1},\mathbf{1},1,0) \longrightarrow (1,1,0,1) \longrightarrow (1,1,0,1) \longrightarrow (1,\mathbf{1},1,\mathbf{0})\end{aligned}$$

5. Constructing a quantum Toffoli gate.

(a) The unitary operation the circuit implement is

$$M = H_3 \cdot T_3 \cdot CNOT_{13} \cdot T_3^{\dagger} \cdot CNOT_{23} \cdot T_3 \cdot CNOT_{13} \cdot T_3^{\dagger} \cdot CNOT_{23} \cdot H_3 = \begin{pmatrix}1 &&&&&&&\\ &1&&&&&&\\ &&1&&&&&\\ &&&1&&&&\\ &&&&1&&&\\ &&&&&1&&\\ &&&&&&0&-i\\ &&&&&&-i&0\end{pmatrix}$$

which is almost a Toffoli gate. The unitary transformation this circuit implements is a controlled-controlled-NOT with an extra phase shift of $e^{-i\pi/2}$. A Toffoli gate is simply a controlled-controlled-NOT, without the extra phase shift.

Mathematica code:

```
(*Problem 5*)

H3 = KroneckerProduct[Id, KroneckerProduct[Id, HadamardMatrix[2]]];

T3 = KroneckerProduct[Id,
KroneckerProduct[Id, {{1, 0}, {0, Exp[I*Pi/4]}}]];
```

```
CNOT = {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 0, 1}, {0, 0, 1, 0}};

CNOT12 = KroneckerProduct[CNOT, Id];

CNOT23 = KroneckerProduct[Id, CNOT];

CNOT13 = KroneckerProduct[SWAP12, Id] . CNOT23 .
KroneckerProduct[SWAP12, Id];

In[121]:= circ =
H3 . T3 . CNOT13 . ConjugateTranspose[T3] . CNOT23 . T3 . CNOT13 .
ConjugateTranspose[T3] . CNOT23 . H3

Out[121]= {{1, 0, 0, 0, 0, 0, 0, 0}, {0, 1, 0, 0, 0, 0, 0, 0}, {0, 0,
         1, 0, 0, 0, 0, 0}, {0, 0, 0, 1, 0, 0, 0, 0}, {0, 0, 0, 0, 1, 0, 0,
         0}, {0, 0, 0, 0, 0, 1, 0, 0}, {0, 0, 0, 0, 0, 0, 0, -I}, {0, 0, 0,
         0, 0, 0, -I, 0}}
```

(b) To correct for the undesired phase shift, we also have to apply some sort of phase shift on the first two quantum wires as well. However, since we don't want to add the phase shift for all inputs, it has to be a controlled-phase shift gate. Consider the ansatz:

$$CP_{12}(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\phi} \end{pmatrix}$$

Apply this after the circuit given by the problem, we find

$$(CP_{12}(\phi) \otimes \mathbb{I}_3) \cdot M = \begin{pmatrix} 1 &&&&&&& \\ & 1 &&&&&& \\ && 1 &&&&& \\ &&& 1 &&&& \\ &&&& 1 &&& \\ &&&&& 1 && \\ &&&&&& 0 & -ie^{i\phi} \\ &&&&&& -ie^{i\phi} & 0 \end{pmatrix}$$

We see that to get the desired Toffoli gate, we want to set $\phi = \pi/2$. This means that we want to construct the gate

$$CP_{12}(\pi/2) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & i \end{pmatrix}$$

To this end, we will need CNOTs (both of them!) and the single-qubit phase gate

$$P(\theta) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

We observe that $CP_{12}(\pi/2)$ is close to a controlled-$R_z(\pi/2)$, but not quite. So, we start by decomposing $CP_{12}(\pi/2)$ as

$$CP_{12}(\pi/2) = \begin{pmatrix} 1 &&& \\ & 1 && \\ && e^{i\pi/4} & \\ &&& e^{i\pi/4} \end{pmatrix} \begin{pmatrix} 1 &&& \\ & 1 && \\ && e^{-i\pi/4} & \\ &&& e^{i\pi/4} \end{pmatrix} = M_1 \cdot M_2$$

M_2 is simply a controlled-$R_z(\pi/2)$, which can be constructed through the procedure given in lecture

$$M_2 = \mathbb{I}_1 \otimes R_z(-\pi/4) \cdot CNOT_{12} \cdot \mathbb{I}_1 \otimes R_z(\pi/4) \cdot CNOT_{12}$$

where

$$R_z(\theta) = \begin{pmatrix} e^{i\theta/2} & 0 \\ 0 & e^{-i\theta/2} \end{pmatrix}.$$

M_1 is given by

$$\begin{aligned} M_1 &= SWAP \cdot (\mathbb{I}_1 \otimes P(\pi/4)) \cdot SWAP \\ &= CNOT_{12}\, CNOT_{21}\, CNOT_{12} \cdot \mathbb{I}_1 \otimes P(\pi/4) \cdot CNOT_{12}\, CNOT_{21}\, CNOT_{12}. \end{aligned}$$

Here I'm following $CNOT_{12} = CNOT_1$ and $CNOT_{21} = CNOT_2$, with $CNOT_i$ defined as in Problem 2. With this, we're done. The Toffoli gate can now be constructed from the circuit provided by the problem and single qubit gates and CNOTs on the first two quantum wires:

$$\begin{aligned} \text{Toff} &= (CP_{12}(\pi/2) \otimes \mathbb{I}_3) \cdot M \\ &= [(M_1 \cdot M_2) \otimes \mathbb{I}_3] \cdot M \\ &= [(C_{12}\, C_{21}\, C_{12} \cdot \mathbb{I}_1 \otimes P(\pi/4) \cdot C_{12}\, C_{21}\, C_{12} \cdot \mathbb{I}_1 \otimes R_z(-\pi/4) \cdot C_{12} \cdot \mathbb{I}_1 \otimes R_z(\pi/4) \cdot C_{12}) \otimes \mathbb{I}_3] \cdot M \end{aligned}$$

where I've abbreviated $CNOT_i$ with C_i.

It appears that we're using a lot of gates to accomplish something that seems very simple. It turns out that we are indeed. Following the decomposition rule given on page 181 of Nielsen and Chuang's, we want to find matrices A, B, C and some angle α for which

$$P(\pi/2) = e^{i\alpha} A \sigma_x B \sigma_x C \qquad ABC = \mathbb{I}.$$

By hook or by crook, one may find the following solution:

$$\alpha = \frac{\pi}{4}, \qquad A = \mathbb{I}, \qquad B = C = \begin{pmatrix} 0 & e^{+i\pi/8} \\ e^{-i\pi/8} & 0 \end{pmatrix}.$$

And the desired circuit takes the form:

$$(P(\pi/4) \otimes \mathbb{I}) \cdot CNOT_{12} \cdot (\mathbb{I} \otimes B) \cdot CNOT_{12} \cdot (\mathbb{I} \otimes C)$$

Calculating the expression above, we find that it is equal to

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & i \end{pmatrix}$$

which is equal to $CP_{12}(\pi/2)$, as desired. Notice that we use way fewer gates than before.

Mathematica code:

```
In[18]:= b = {{0, Exp[I*Pi/8]}, {Exp[-I*Pi/8], 0}};

In[19]:= \[Sigma]x = PauliMatrix[1];

In[20]:= \[Sigma]x . b

Out[20]= {{E^(-((I \[Pi])/8)), 0}, {0, E^((I \[Pi])/8)}}

In[24]:= KroneckerProduct[P[Pi/4], Id] . CNOT .
KroneckerProduct[Id, b] . CNOT . KroneckerProduct[Id, b]

Out[24]= {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, I}}
```