2. In this question you will construct and compare quantum programs in different formats.

   (a) Recall the following standard quantum gates:

$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \qquad H = \tfrac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\mathrm{CX} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad \mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

   Show that $S^2 = Z$ and $(H \otimes H) \circ \mathrm{CX} \circ (H \otimes H) = \mathrm{SWAP} \circ \mathrm{CX} \circ \mathrm{SWAP}$. [*3 marks*]

   (b) Each of the five small Qiskit programs `a-e` below matches one of the pieces of data A-E. Which goes with which?

   [*5 marks*]

```
a = qs.QuantumCircuit(2)
a.s(0)
```

```
b = qs.QuantumCircuit(2)
b.cx(0,1)
b.x(0)
```

```
c = qs.QuantumCircuit(2)
c.x(0)
c.x(1)
```

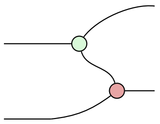
```
d = qs.QuantumCircuit(2)
d.x(0)
d.cx(0,1)
```

```
e = qs.QuantumCircuit(2)
```

A:

```
op1 :: Qubit -> Qubit
    -> Circ (Qubit, Qubit)
op1 q0 q1 = do
q0’ <- qnot q0
q1’ <- qnot q1
return (qo’,q1’)
```

C:



B:

```
op2 :: Qubit -> Qubit
    -> Circ (Qubit, Qubit)
op2 q0 q1 = return (qo,q1)
```

D:

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

E:

```
include "stdgates.inc";
qubit[2] q;
ctrl @ x q[0],q[1];
x q[0];
```

The qiskit code below specifies three quantum circuits (1–3) that have one *hole* each in it. This hole acts as a placeholder. A *complete* circuit is constructed by plugging one of `a-e` into the hole of one of the circuits (1–3).

```
import qiskit as qs

qc1 = qs.QuantumCircuit(2)
qc1.h(1)
qc1.cx(1,0)
qc1.s(0)
qc1.h(1)

qc2 = qs.QuantumCircuit(2)
qc2.h(0)

qc3.= qs.QuantumCircuit(2)
qc3.cx(0,1)

# placeholder circuit
hole = qs.QuantumCircuit(2)

circuit1 = qc1.compose(hole).compose(qc2)

circuit2 = qc2.compose(hole)

circuit3 = qc2.compose(hole).compose(qc3)
```

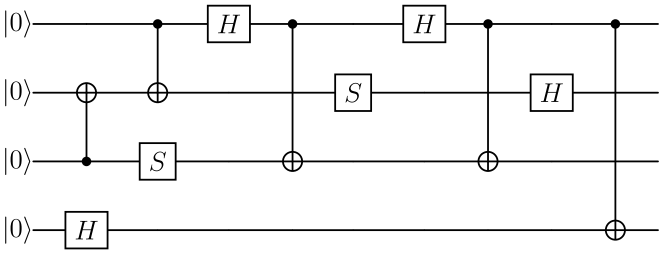
(c) Prove that the complete circuits `circuit1.a` and `circuit2.a` are semantically different. [*3 marks*]

(d) Which two pairs of the possible complete circuits are semantically equivalent, and why? [*8 marks*]

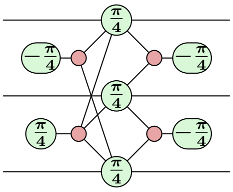
(e) Name three (different) pairs of complete circuits that you can make semantically equivalent by adding a single X gate to one of them, and explain why. [*6 marks*]

3. This question applies ZX calculus to quantum circuits.

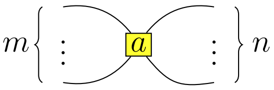
   (a) Write the following circuit with input states as a ZX-diagram and simplify it. Which qubits are entangled to each other? *[5 marks]*



   (b) Compute the semantic interpretation as a matrix to verify that this ZX-diagram implements a CCZ gate: *[5 marks]*



The rest of the question uses the following *H-boxes*:

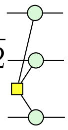


$$= \frac{1}{\sqrt{2}}\sum a^{i_1\ldots i_m j_1\ldots j_n}|j_1\ldots j_n\rangle\langle i_1\ldots i_m|$$

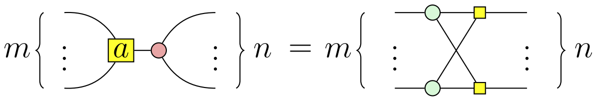
The sum ranges over all $i_1,\ldots,i_m,j_1,\ldots,j_n \in \{0,1\}$, and a is an arbitrary complex number. Hence an H-box represents a matrix with, up to a global factor of $\sqrt{2}$, all entries equal to 1, except for the bottom right entry, which is equal to a. In particular, if $a = -1$ and $m = n = 2$, it is just a Hadamard gate, and we'll leave the label $a = -1$ out of the picture.

   (c) Compute the semantic interpretation as a matrix to verify that this ZX-diagram also implements the CCZ gate: *[5 marks]*

$$\text{CCZ} = \sqrt{2}$$



   (d) Prove that H-boxes satisfy the following equation. *[5 marks]*



   (e) Use (d) to prove using ZX-calculus that CCZ is its own inverse. *[5 marks]*