Performing operations with Operator and State vector

Unitary operations can be defined and performed on state vectors in Qiskit using the Operator class

```
import qiskit
from qiskit.quantum_info import Operator
from qiskit.quantum_info import Statevector
from numpy import sqrt

X = Operator([[0, 1], [1, 0]])
Y = Operator([[0, -1.0j], [1.0j, 0]])
Z = Operator([[1, 0], [0, -1]])
H = Operator([[1 / sqrt(2), 1 / sqrt(2)], [1 / sqrt(2), -1 / sqrt(2)]])
S = Operator([[1, 0], [0, 1.0j]])
T = Operator([[1, 0], [0, (1 + 1.0j) / sqrt(2)]])

v = Statevector([1, 0])
```

```
v = v.evolve(H)
v = v.evolve(T)
v = v.evolve(H)
v = v.evolve(T)
v = v.evolve(Z)

display(v.draw("text"))
display(v.draw("latex"))
```

[1] ✓ 3.2s Python

```
[ 0.85355339+0.35355339j,-0.35355339+0.14644661j]
```

$$(0.8535533906 + 0.3535533906i)|0\rangle + (-0.3535533906 + 0.1464466094i)|1\rangle$$

Quantum circuits

Qiskit's Quantum Circuit class. In particular, we may define a quantum circuit (which in this case will simply be a sequence of unitary operations performed on a single qubit) as follows.

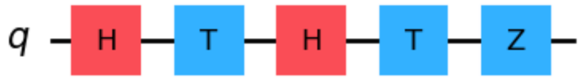
```python
from qiskit import QuantumCircuit

circuit = QuantumCircuit(1)

circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.t(0)
circuit.z(0)

display(circuit.draw(output='mpl'))
```

[4] Python



The operations are applied sequentially, starting on the left and ending on the right in the figure. Let us first initialize a starting quantum state vector and then evolve that state according to the sequence of operations.
Finally, let's simulate the result of running this experiment (i.e., preparing the state $|0\rangle$ applying the sequence of operations represented by the circuit, and measuring) 4000 times.

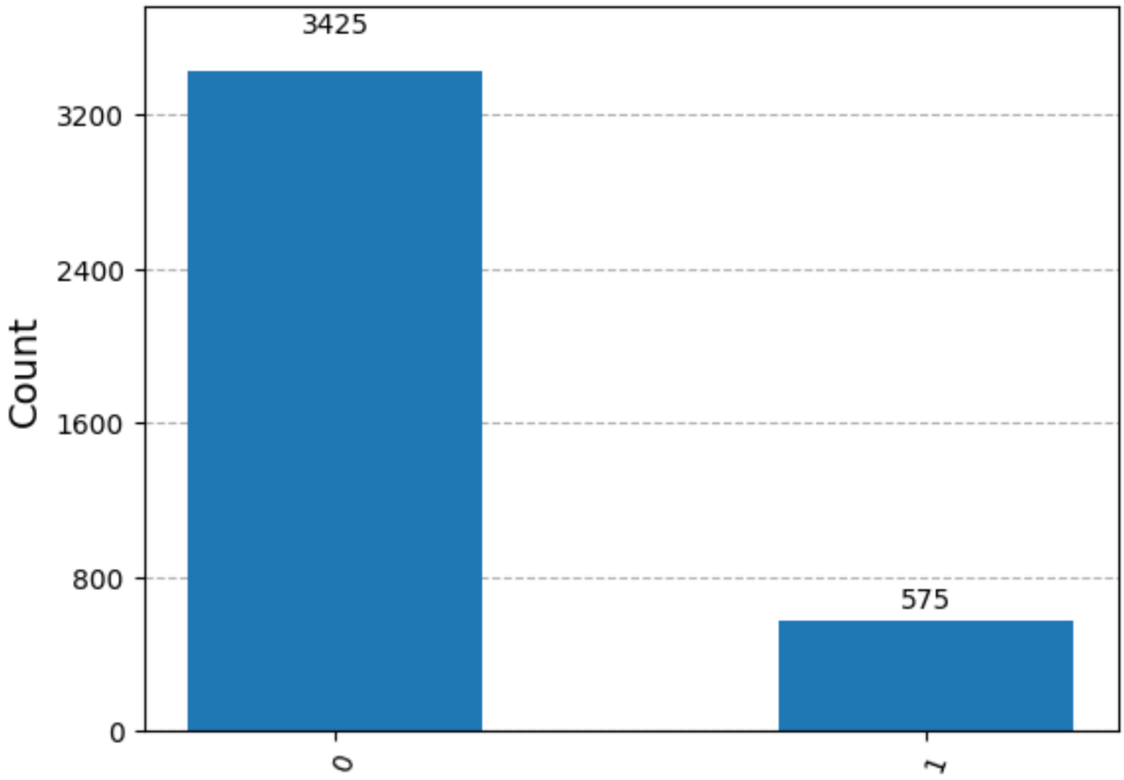
```python
ket0 = Statevector([1, 0])
v = ket0.evolve(circuit)
v.draw("text")
```

[4] Python

```
[ 0.85355339+0.35355339j,-0.35355339+0.14644661j]
```

```python
from qiskit.visualization import plot_histogram
statistics = v.sample_counts(4000)
plot_histogram(statistics)
```

[7] Python



3425
575
3200
2400
1600
800
0
Count
0
1