

Jupyter Notebook Qiskit Code and Simulations

```
import numpy as np
from qiskit import QuantumCircuit
from qiskit.quantum_info import Operator
from qiskit import *
from qiskit.visualization import plot_histogram
from qiskit.tools.monitor import job_monitor
```

```
quanCrct = QuantumCircuit(4, 4)
```

```
quanCrct.h(0) # Adding H gate to first qubit (qubit 0). This creates a
superposition on this qubit
```

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f089a67c0>
```

```
quanCrct.cx(0,1) # Add a CNOT gate on qubits 0 (control) and 1 (target)
creating an entanglement (Bell state)
```

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f08463ec0>
```

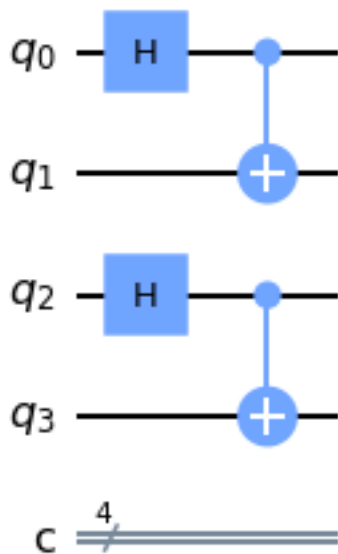
```
quanCrct.h(2) # Add H gate to qubit 2, superposition
```

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f089af9c0>
```

```
quanCrct.cx(2,3) # Add CNOT Gate on qubit 2 and 3 (control qubit 2 and target
3)
```

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f089b53c0>
```

```
quanCrct.draw('mpl') # Draw circuit with matplotlib
```



In [11]:

```
U = Operator(quanCrct)
U.data
```

Out[11]:

```
array([[ 0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j,  0.5+0.j,  0.5+0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j],
       [ 0. +0.j,  0. +0.j,  0.5+0.j, -0.5+0.j,  0. +0.j,  0. +0.j,
         0.5+0.j, -0.5+0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j],
       [ 0. +0.j,  0. +0.j,  0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j,
         0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j],
       [ 0.5+0.j, -0.5+0.j,  0. +0.j,  0. +0.j,  0.5+0.j, -0.5+0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j],
       [ 0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j,
        -0.5+0.j, -0.5+0.j,  0. +0.j,  0. +0.j],
       [ 0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0.5+0.j, -0.5+0.j,
         0. +0.j,  0. +0.j, -0.5+0.j,  0.5+0.j],
       [ 0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0.5+0.j,  0.5+0.j,
         0. +0.j,  0. +0.j, -0.5+0.j, -0.5+0.j],
       [ 0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0.5+0.j, -0.5+0.j,  0. +0.j,  0. +0.j,
        -0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j],
       [ 0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j,
         0.5+0.j,  0.5+0.j,  0. +0.j,  0. +0.j],
       [ 0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j,  0.5+0.j, -0.5+0.j,
         0. +0.j,  0. +0.j,  0. +0.j,  0. +0.j]])
```

```

    0. +0.j, 0. +0.j, 0.5+0.j, -0.5+0.j],
[ 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0.5+0.j, 0.5+0.j,
 0. +0.j, 0. +0.j, 0.5+0.j, 0.5+0.j],
[ 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j,
 0. +0.j, 0. +0.j, 0.5+0.j, -0.5+0.j, 0. +0.j, 0. +0.j,
 0.5+0.j, -0.5+0.j, 0. +0.j, 0. +0.j],
[ 0.5+0.j, 0.5+0.j, 0. +0.j, 0. +0.j, -0.5+0.j, -0.5+0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j],
[ 0. +0.j, 0. +0.j, 0.5+0.j, -0.5+0.j, 0. +0.j, 0. +0.j,
-0.5+0.j, 0.5+0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j],
[ 0. +0.j, 0. +0.j, 0.5+0.j, 0.5+0.j, 0. +0.j, 0. +0.j,
-0.5+0.j, -0.5+0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j],
[ 0.5+0.j, -0.5+0.j, 0. +0.j, 0. +0.j, -0.5+0.j, 0.5+0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j,
 0. +0.j, 0. +0.j, 0. +0.j, 0. +0.j]]))

```

In [12]:

```
quanCrct.h(2)
```

Out[12]:

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f095d5fc0>
```

In [13]:

```
quanCrct.h(3)
```

Out[13]:

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f0a6253c0>
```

In [14]:

```
quanCrct.y(2)
```

Out[14]:

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f0a6476c0>
```

In [15]:

```
quanCrct.z(3)
```

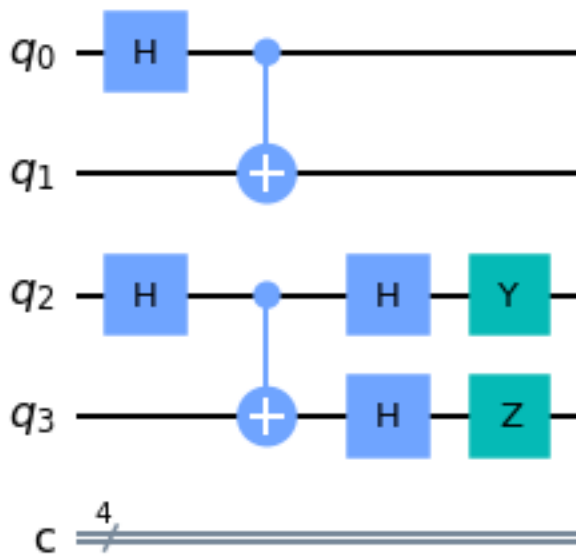
Out[15]:

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f09590a80>
```

In [16]:

```
quanCrct.draw('mpl') # Draw circuit with matplotlib
```

Out[16]:



In [17]:

```
U = Operator(quanCrct)
U.data
```

Out[17]:

```
array([[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j ,
        0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j,
        0.+0.j , 0.+0.j ],
       [0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
        0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ],
       [0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
        0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
        0.-0.5j, 0.-0.5j],
       [0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j ,
        0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j,
        0.+0.j , 0.+0.j ],
       [0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ],
       [0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ],
       [0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ],
       [0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ],
       [0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ],
       [0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j , 0.+0.j ,
        0.+0.j , 0.+0.j ]]
```

```

0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.5j, 0.-0.5j],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.5j, 0.+0.5j],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j,
0.+0.j , 0.+0.j ]])

```

In [18]:

```
quanCrct.cx(1,2)
```

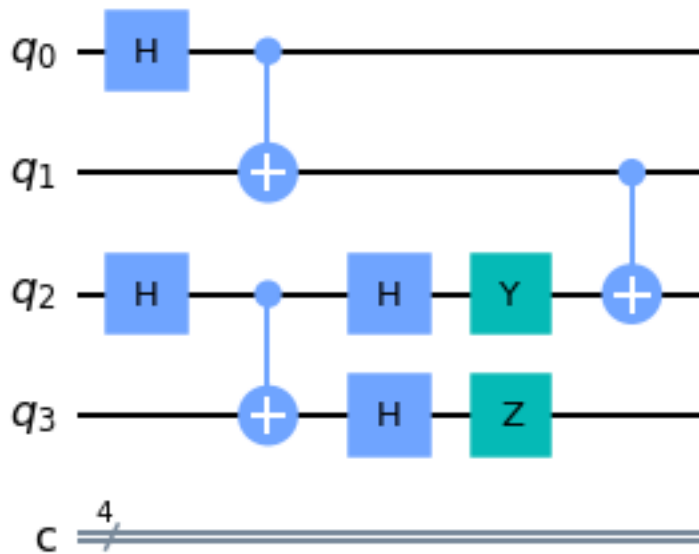
Out[18]:

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f0a6e8540>
```

In [19]:

```
quanCrct.draw('mpl')
```

Out[19]:



In [20]:

```
U = Operator(quanCrct)
U.data
```

Out[20]:

```
array([[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j ,
```

```

0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.-0.5j, 0.+0.5j],
[0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.-0.5j, 0.-0.5j],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j,
0.+0.j , 0.+0.j ],
[0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.5j, 0.+0.5j],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.-0.5j,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j,
0.+0.j , 0.+0.j ],
[0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j,
0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.5j, 0.-0.5j],
[0.+0.j , 0.+0.j , 0.+0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j , 0.-0.5j, 0.-0.5j, 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.5j, 0.-0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.-0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j ],
[0.+0.5j, 0.-0.5j, 0.+0.5j, 0.+0.j , 0.+0.j , 0.+0.j , 0.+0.j ,
0.+0.j , 0.+0.j , 0.+0.j ]])

```

In [21]:

```

meas = QuantumCircuit(4, 4)
meas.measure([0,1], [0,1])

```

Out[21]:

```
<qiskit.circuit.instructionset.InstructionSet at 0x29f0a822c40>
```

In [22]:

```
backend = BasicAer.get_backend('qasm_simulator')
```

In [23]:

```
circ = quanCrct.compose(meas)
```

In [24]:

```
result = backend.run(transpile(circ, backend), shots=1000).result()
```

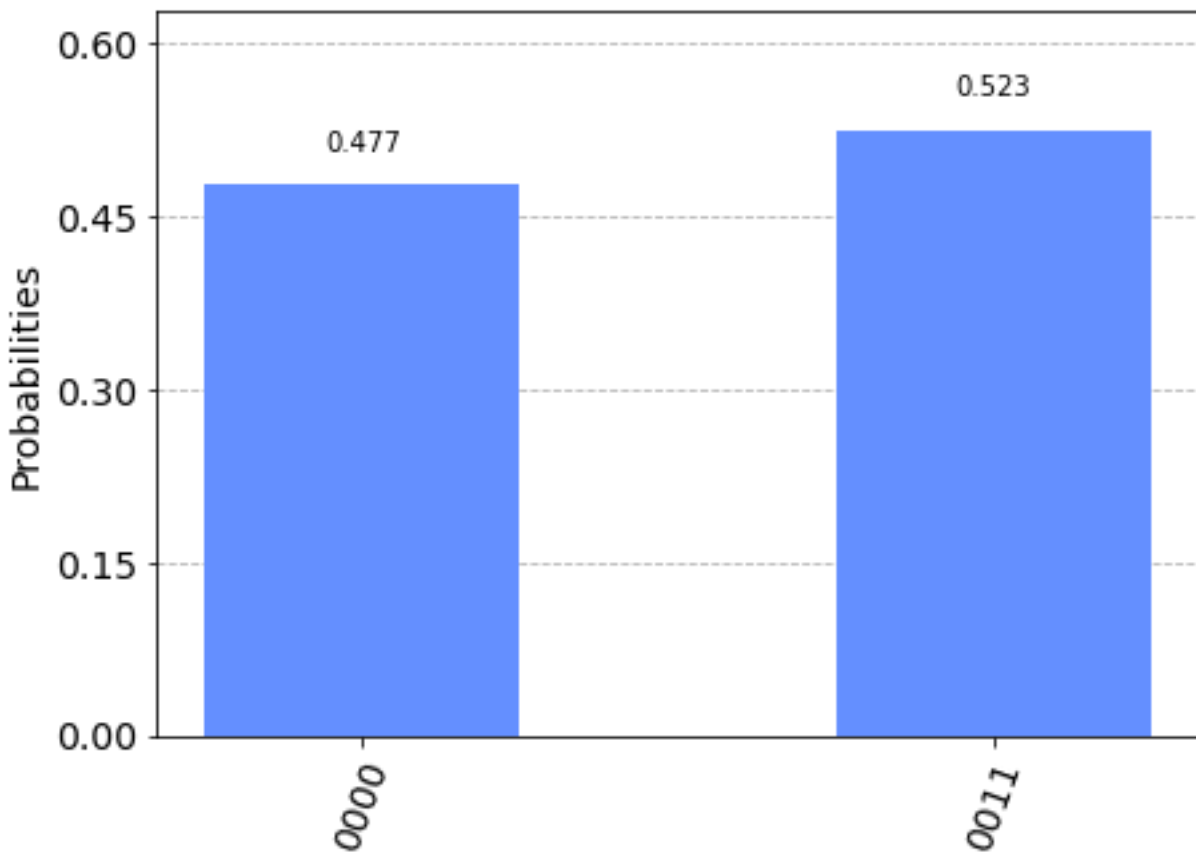
In [25]:

```
counts = result.get_counts(quanCrct)
print(counts)
{'0000': 477, '0011': 523}
```

In [26]:

```
plot_histogram(counts)
```

Out[26]:



In []: