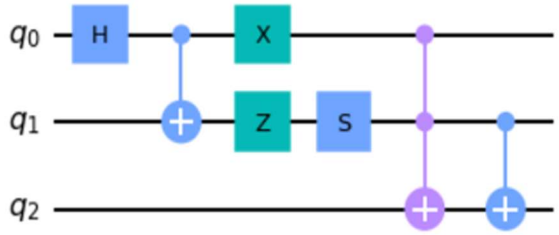
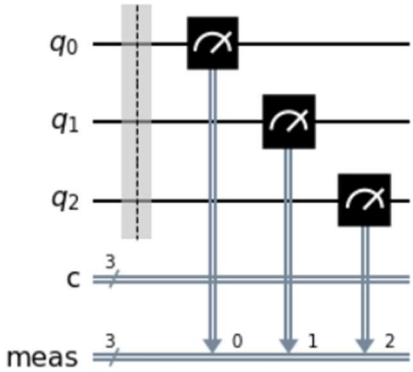
1) What is the depth of the given circuit?



a) 4
b) 5
c) 6
d) 7

2) Given an empty QuantumCircuit object, qc, with three qubits and three classical bits, which one of these code fragments would create this circuit?



a) qc.measure([0,1,2],[0,1,2])
b) qc.barrier()
qc.measure([0,1,2],[0,1,2])
c) qc.measure_all()
d) qc.measure([0,0],[1,1],[2,2])

3) Which of the following code fragments would create a quantum circuit with 4 quantum bits and 2 classical bits?

a) qc = QuantumCircuit(2,4)
b) q = QuantumRegister(2)
c = ClassicalRegister(4)
qc = QuantumCircuit(q,c)
c) qc = QuantumCircuit([4,2])
d) qc = QuantumCircuit(4,2)

4) Given this code fragment, what is the probability that a measurement would result in |1>?

```
qc = QuantumCircuit(1)
qc.h(0)
qc.ry(pi/4,0)
```

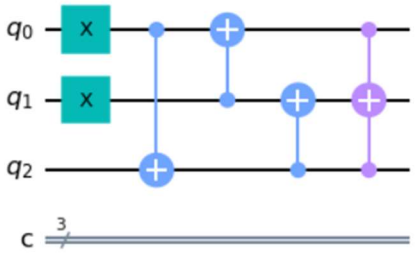
a) 0.137

b) 0.863
c) 0.5
d) 1.0

5) What is the output after executing the following circuit?

a) 11 and 00
b) 01 and 10
c) 11 and 01
d) 10 and 11

6) Assuming the fragment below, which three code fragments would produce the circuit illustrated?



a)
```
qc = QuantumCircuit(3,3)
qc.x([0,1])
qc.cx(1,0)
qc.cx(0,2)
qc.cx(2,1)
qc.ccx(0,1,2)
```

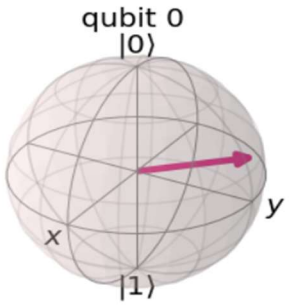
b)
```
qc = QuantumCircuit(3,3)
for i in range(3):
    qc.x(i)
for j in range(3):
    qc.cx(j-1,j)
qc.ccx(0,1,2)
```

c)
```
qc = QuantumCircuit(3,3)
for i in range(2):
    qc.x(i)
for j in range(3):
    qc.cx(j,j-1)
qc.ccx(0,2,1)
```

d) qc = QuantumCircuit(3,3)
qc.x([0,1])
qc.cx(0,1)
qc.cx(0,2)
qc.cx(2,1)
qc.ccx(0,2,1)

7) Given this code, which two inserted code fragments result in the state vector represented by this Bloch sphere?

```
qc= QuantumCircuit(1,1)
# Insert code fragment here
simulator = Aer.get_backend('statevector_simulator')
job = execute(qc, simulator)
result = job.result()
outputstate = result.get_statevector(qc)
plot_bloch_multivector(outputstate)
```



a) qc.x(0)
qc.s(0)
qc.t(0)

b) qc.h(0)
qc.sdg(0)
qc.z(0)
qc.t(0)

c) qc.h(0)
qc.s(0)
qc.x(0)

d) qc.sdg(0)
qc.h(0)
qc.x(0)
qc.t(0)

8) Which of the following gates can be used to build any single qubit gate?

a) U gate
b) Z gate
c) P gate

d) I gate
e) All of the above

9) Which of the following given options would create a barrier across all qubits?

a) qc.barrier_all()
b) qc.barrier()
c) qc.get_barrier()
d) qc.barrier.get_all()

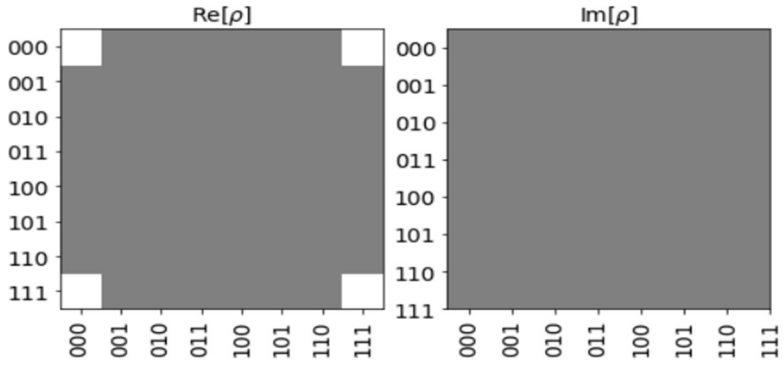
10) Which one of the following codes will result in an Z gate?

a) HSH
b) HXH
c) HHH
d) ZXZ

11) Which of the following codes given below creates a perfect bell state?

a)
```
bell = QuantumCircuit(2)
bell.h(0)
bell.cx(0,1)
```
b)
```
bell = QuantumCircuit(2)
bell.h(0)
bell.z(0)
bell.cx(0,1)
```
c)
```
bell = QuantumCircuit(2)
bell.h(0)
bell.x(1)
bell.cx(0,1)
```
d)
```
bell = QuantumCircuit(2)
bell.h(0)
bell.x(1)
bell.cx(0,1)
bell.z(1)
```

12) Given the Hinton diagram of the GHZ state, Choose the correct code that represents the given state.

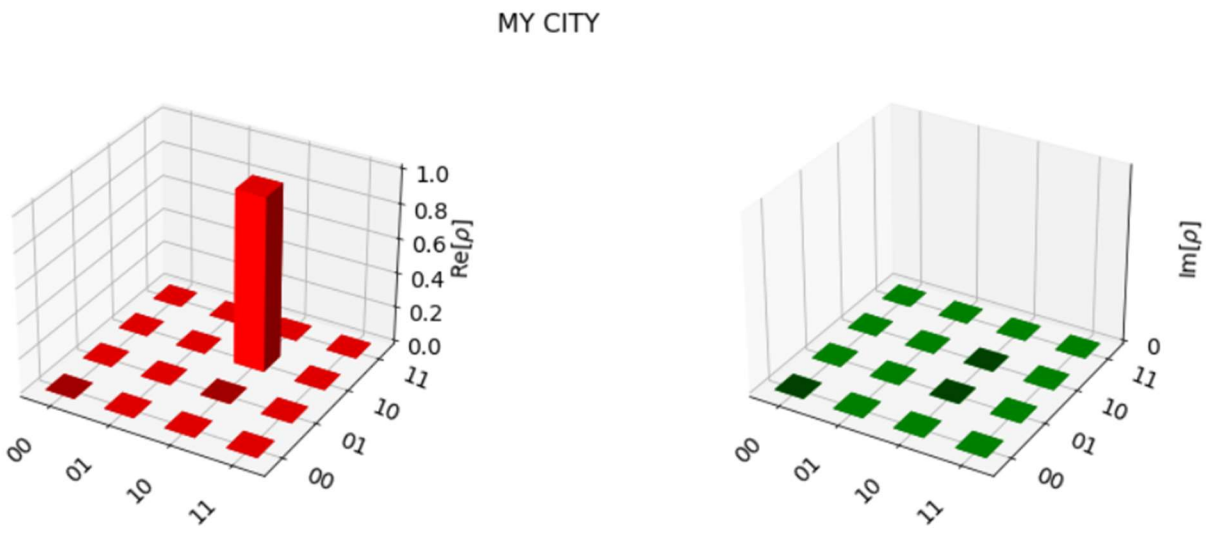


a)
```
qc = QuantumCircuit(3)
qc.h(0)
qc.cx([0,0],[1,2])
```
b)
```
qc = QuantumCircuit(3)
qc.x(0)
qc.cx([0,0],[1,2])
```

c) qc = QuantumCircuit(3)
qc.h(0)
qc.z(0)
qc.cx([0,0],[1,2])

d) qc = QuantumCircuit(3)
qc.h(0)
qc.cx([0,1],[2,3])

13) Given the city diagram, Choose the correct code fragment when executed gives the figure given.



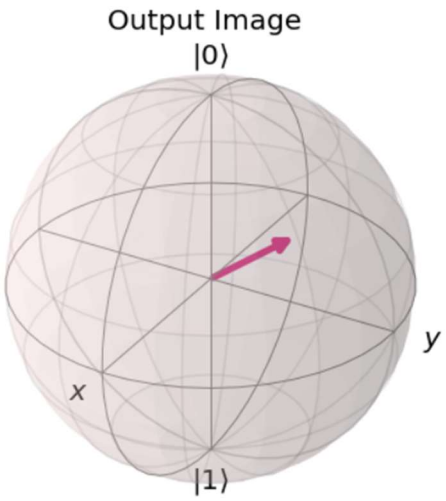
a) qc = QuantumCircuit(3)
qc.x(0)
qc.cx(0,2)
qc.cx(1,0)

b) qc = QuantumCircuit(2)
qc.h(0)
qc.cx(0,1)
qc.cx(2,0)

c) qc = QuantumCircuit(3)
qc.x(0)
qc.cx(2,1)
qc.cx(1,0)

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

14) Which one of the following codes given below will result in the image shown?



a) cords=[1,1,1]
plot_bloch_vector(cords, title='Image')
b) cords=[0,1,0]
plot_bloch_vector(cords, name='Image')
c) cords=[1,0,0]
plot_bloch_vector(cords, title='Image')
d) cords=[0,0,1]
plot_bloch_vector(cords, name='Image')

15) Which code snippet would execute a circuit given these parameters?

- Use the statevector Simulator
- Measure the circuit 1000 times
- Use a coupling map that connects three qubits linearly
- Convert the output to latex

```
qc = QuantumCircuit(3)
# Insert code fragment here
result = job.result()
```

a)
```
backend = Aer.get_backend('statevector_simulator')
couple_map = [[0, 1], [1, 2]]
job = execute(qc, backend, shot=1024, coupling_map=couple_map)
output = result.get_statevector()
statevector = array.get_latex(output)
```

b)
```
backend = Aer.get_backend('statevector_simulator')
couple_map = [[0, 1], [1, 2]]
job = execute(qc, backend, loop=1024, coupling_map=couple_map)
output = result.get_statevector()
statevector = array.get_latex(output)
```

c)
```
backend = Aer.get_backend('statevector_simulator')
couple_map = [[0, 1], [1, 2]]
job = execute(qc, backend, shots=1024, coupling_map=couple_map)
output = result.get_statevector()
```

statevector = array_to_latex(output)

d) backend = Aer.get_backend('statevector_simulator')
couple_map = [[0, 1], [1, 2]]
job = execute(qc, backend, shot=1024, coupling_map=couple_map)
output = result.get_statevector()
statevector = array_to_latex(output)

16) Which one of the following codes will calculate the state fidelity of the given code below?

initial_state = [1/sqrt(2), -1/sqrt(2)]
alpha = random_statevector(2)
state1 = initial_state
state2 = initial_state
# Insert your code here

a) result.get_state.fidelity(state1, state2)
b) get_state.fidelity(state1, state2)
c) state_fidelity(state1, state2)
d) fidelity.get_state(state1, state2)

17) Which code fragment would yield an operator that gives the given output?

```
Operator([[0.+0.j, 1.+0.j, 0.+0.j, 0.+0.j],
          [0.+0.j, 0.+0.j, 1.+0.j, 0.+0.j],
          [0.+0.j, 0.+0.j, 0.+0.j, 1.+0.j],
          [1.+0.j, 0.+0.j, 0.+0.j, 0.+0.j]],
         input_dims=(2, 2), output_dims=(2, 2))
```

a) qc = QuantumCircuit(2)
qc.x(0)
qc.cx(0,1)
op = Operator(qc)
b) qc = QuantumCircuit(3)
qc.x(0)
qc.cx(1,2)
op = Operator(qc)
c) op = Operator(op.x(0))
op.cx(0,1)
d) op = Operator.Xop(0)
op.cx(0,1)

18) Which one of the following codes are used to compose A.B?

A = Operator(Pauli(label='X'))
B = Operator(Pauli(label='Z'))
# Insert your code here

a) A.compose(B)
b) A.get_compose(B)
c) A.B
d) A.expand(B)

19) Which two of the given codes below are False (or) result False when executed.

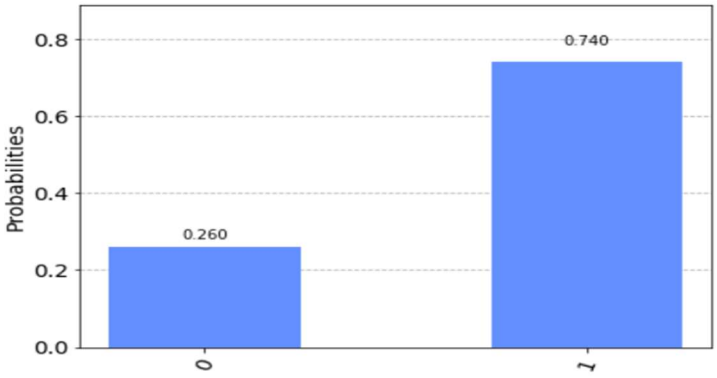
a) Operator(Pauli(label='X')) == Operator(XGate())
b) Operator(XGate()) == np.exp(1j * 0.5) * Operator(XGate())
c) Operator(Pauli(name='X')) == Operator(XGate())
d) Operator(XGate()) == np.exp(1j * 0) * Operator(XGate())

20) Which one of the following output results are correct when the given code is executed?

```
qc = QuantumCircuit(2)
qc.h(0)
qc.h(1)
backend = Aer.get_backend('statevector_simulator')
job = execute(qc, backend)
statevector = job.result().get_statevector()
print(statevector)
```

a) [0.5+0.j, 0.5+0.j, 0.5+0.j, 0.5+0.j]
b) [1.+0.j, 0.+0.j, 0.+0.j, 0.+0.j]
c) [0.70710678+0.00000000e+00j, -0.70710678+8.65956056e-17j,
0. +0.00000000e+00j, 0. +0.00000000e+00j]
d) [0.70710678+0.j, 0.70710678+0.j, 0. +0.j, 0. +0.j]

21) Which of the following codes given below when executed gives the histogram result as shown in the image given below.

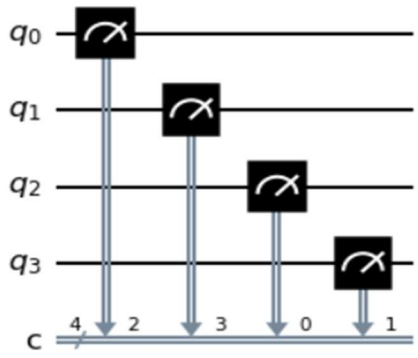


a) qc.ry(pi/3,0)
qc.x(0)
b) qc.x(0)
qc.ry(3*pi/6,0)
c) qc.h(0)
qc.x(0)
d) qc.ry(pi/2,0)
qc.h(0)

22) Ry gate with an angle of rotation 90 degrees can be expressed using U gate as?

a) U(0,pi/2,0)
b) U(pi/2,0,0)
c) U(0,0,pi/2)
d) All of the above

23) Given an empty QuantumCircuit object, qc, with four qubits and four classical bits, which two of these code fragments would create the circuit shown below?



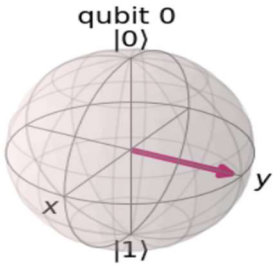
a) qc.measure(range(4),range(4))
b) qc.measure([0,1,2,3],[0,1,2,3])
c) qc.measure_all()
d) qc.measure([0,1,2,3],[2,3,0,1,])
e) for i in range(4):
   qc.measure(i,i-2)

24) S-gate is a Qiskit phase gate with what value of the phase parameter?

a) pi/2
b) pi/4
c) pi/6
d) 3*pi/4

25) When executed, which two of the following codes produce the multi-vector of the image given below?

```
qc = QuantumCircuit(1)
#Insert your code here
backend = Aer.get_backend('statevector_simulator')
result = execute(qc, backend).result()
statevector = result.get_statevector()
plot_bloch_multivector(statevector)
```



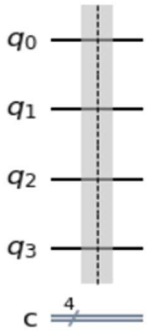
a) qc.ry(-pi,0)
   qc.rx(pi/2,0)
b) qc.ry(pi,0)
   qc.ry(-pi/2,0)
c) qc.ry(pi,0)

qc.rx(-pi/2,0)

d) qc.rx(-pi,0)
qc.rx(-pi/2,0)

e) qc.ry(pi,0)
qc.rx(pi/2,0)

26) Which three of the following codes given when executed results in the output image shown below.



a) qc.barrier_all()
b) qc.barrier([0,1,2,3])
c) qc.barrier(range(4),range(4))
d) for i in range(4):
qc.barrier(i)
e) qc.barrier(range(4))

27) What code fragment gives the equivalent circuit if you remove the barrier in the following Quantum Circuit given below?

a) qc.h(0)
qc.rx(-pi/2,0)

b) qc.h(0)
qc.ry(-pi/2,0)

c) qc.h(0)
qc.rz(-pi/2,0)

d) qc.x(0)
qc.ry(-pi/2,0)

e) qc.y(0)
qc.ry(pi/2,0)

28) Which of the following code fragments when executed doesn't give out any error?

a) execute(qc, basicgates=['u1', 'cx', 'ry'])
b) execute(qc, get_basicgates=['u1', 'cx', 'ry'])
c) execute(qc, basic_gates = ['u1', 'cx', 'ry'])
d) execute(qc, basic.get_gates=['u1', 'cx', 'ry'])

29) Which one of the following output results are correct when the given code is executed?

```
qc = QuantumCircuit(1)
qc.h(0)
qc.x(0)
backend = Aer.get_backend('unitary_simulator')
job = execute(qc, backend)
unitary = job.result().get_unitary()
print(unitary)
```

a) [[0.70710678+0.j -0.70710678+0.j]
 [0.70710678+0.j 0.70710678+0.j]]
b) [[0.70710678+0.00000000e+00j 0.70710678+8.65956056e-17j]
 [-0.70710678+8.65956056e-17j 0.70710678+0.00000000e+00j]]
c) [[1.+0.j 0.+0.j]
 [0.+0.j 1.+0.j]]
d) [[0.70710678+0.00000000e+00j 0.70710678-8.65956056e-17j]
 [0.70710678+0.00000000e+00j -0.70710678+8.65956056e-17j]]

30) Which one of the following codes tells us the error rate or read out error in qubits in a given quantum system?

a) plot_error()
b) plot_gate_map()
c) plot_error_map()
d) plot_gate_error()

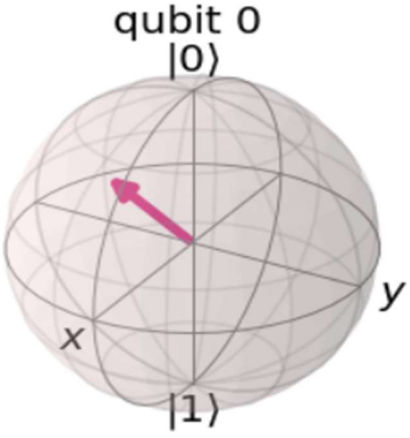
31) Which of the following code fragments given below when executed generates a state vector with dimensions 3x3?

a) random.get_statevector([3,3])
b) get_random.statevector(dims=[3,3])
c) random_statevector(dims=[3,3])
d) statevector.get_random([3,3])

32) Arrange the given code fragments in particular order so that when the arranged code is executed it results in the given Bloch sphere.

```
qc = QuantumCircuit(1)
```

1) qc.rz(pi/3,0)
2) qc.x(0)
3) qc.rx(pi,0)
4) qc.x(0)
5) qc.ry(pi/2,0)



a) 1),5),3),4),2)
b) 2),5),1),4),3)
c) 3),1),2),5),4)
d) 5),1),4),3),2)
e) 4),1),5),4),3)

33) Given a matrix, which one of the following codes given below are used to check whether the given matrix is unitary or not?

```
U = [[1,0,0,0],
     [1,0,0,1],
     [0,1,0,1],
     [1,0,1,0]]
```

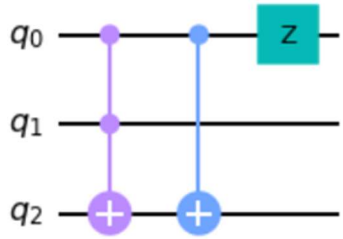
a) U.get_unitary()
b) U.is_unitary()
c) U.unitary()
d) U.check_unitary()

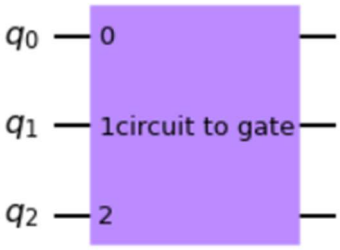
34) Which of the following codes are used to extract existing qasm code named ‘my circuit’ into qiskit code?

a) QuantumCircuit.from_qasm_file(‘my circuit’)
b) QuantumCircuit.get_qasm_file(‘my circuit’)
c) QuantumCircuit.extract_qasm_file(‘my circuit’)
d) QuantumCircuit.to_qiskit_file(‘my circuit’)

35) When the given circuit is wrapped up into a single gate, which one of the following codes given below will help us to do that?

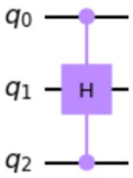
```
qc = QuantumCircuit(3)
qc.mct([0,1],2)
qc.cx(0,2)
qc.z(0)

#Insert your code here
qc_gate.name = 'circuit to gate'
circ = QuantumCircuit(3)
#Insert your code here
```





a) qc_gate = qc.get_gate()
   circ.add(qc_gate, [0,1,2])
b) qc_gate = qc.from_gate()
   circ.add(qc_gate, [0,1,2])
c) qc_gate = qc.to_gate()
   circ.append(qc_gate, [0,1,2])
d) qc_gate = qc.get_gate()
   circ.sum(qc_gate, [0,1,2])

36) Which one of the given code fragments when executed gives the image shown below?



a) ch = HGate().control(3)

qc = QuantumCircuit(3)
qc.append(ch, [0,1,2])

b) ch = HGate().control(3)
qc = QuantumCircuit(3)
qc.append(ch, [0,2,1])

c) ch = HGate().control(2)
qc = QuantumCircuit(3)
qc.append(ch, [0,2,1])

d) ch = HGate().control(1)
qc = QuantumCircuit(3)
qc.append(ch, [0,2,1])

37) What should be the code inserted, so that the original code with given gates gets converted to another code with gates ‘U1’, ‘T’, ‘CX’, but when executed gives the same output.

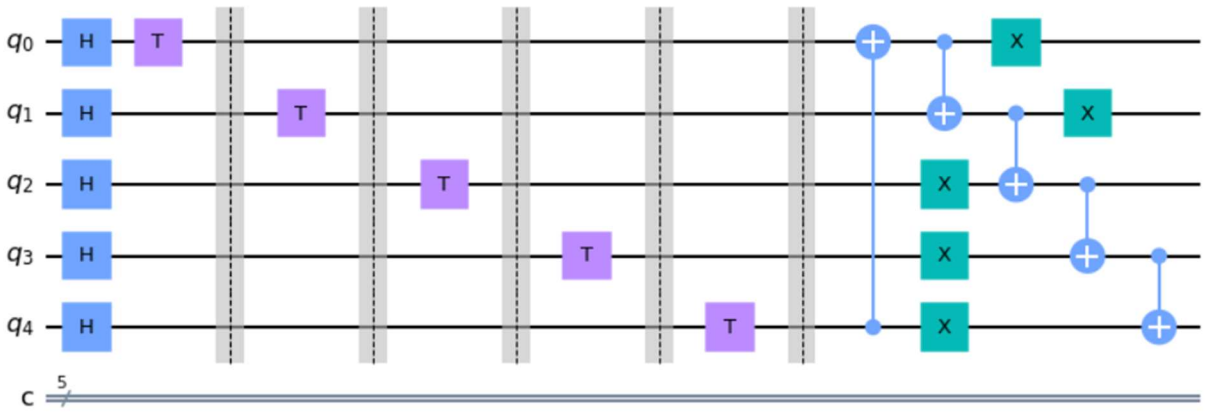
```
qc = QuantumCircuit(3)
qc.mct([0,2],1)
qc.cx(1,2)
qc.h(0)
qc.z(1)
convert = #Insert Your Code Here
convert.draw('mpl')
```

a) convert(qc, basis_gates = ['u1','cx','t'])
b) get_convert(qc, basicgates=['u1','cx','t'])
c) qc.transpile(qc,basic_gates=['u1','cx','t'])
d) transpile(qc, basic_gates=['u1','cx','t'])
e) qc.get_transpile(qc, basicgates=['u1','cx','t'])

38) How many BasicAer simulators are there and how do we check them?

a) 2, BasicAer.get_backends()
b) 4, BasicAer.backends()
c) 3, BasicAer.get_backends()
d) 3, BasicAer.backends()

39) Arrange the given code fragments in particular order so that when the arranged code is executed it results in the given figure below.



1) for i in range(5):

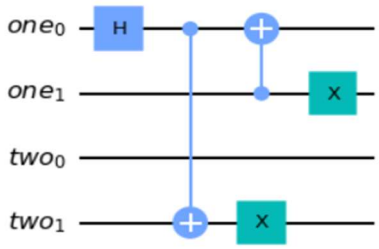
qc.h(i)
2) qc.barrier()
3) for k in range(5):
qc.t(k)
4) for j in range(5):
qc.cx(j-1,j)
5) qc.x(j-3)

a) 1),2),4),5),3)
b) 1),3),2),4),5)
c) 4),2),1),3),5)
d) 3),2),5),4),1)
e) 1),4),3),2),5)
f) 2),5),3),4),1)

40) Which one of the following codes are used to check all the available simulators in ibm-q hub?
provider = IBMQ.get_provider(hub='ibm-q')

a) provider.backends()
b) provider.get_backends()
c) backends.get_provider()
d) backends.get_all('provider')

41) Assuming the figure given below, which one code fragments would produce the circuit illustrated?
inp = QuantumRegister(2, name='one')
out = QuantumRegister(2, name='two')
qc = QuantumCircuit(inp, out)



a) qc.h(0)
qc.cx(0,1)
qc.cx(0,5)
qc.x([0,1])

b) qc.h(inp[0])
for i in range(2):
qc.cx(i,i-1)
qc.x([1,3])

c) qc.h(out[0])
qc.cx(0,5)

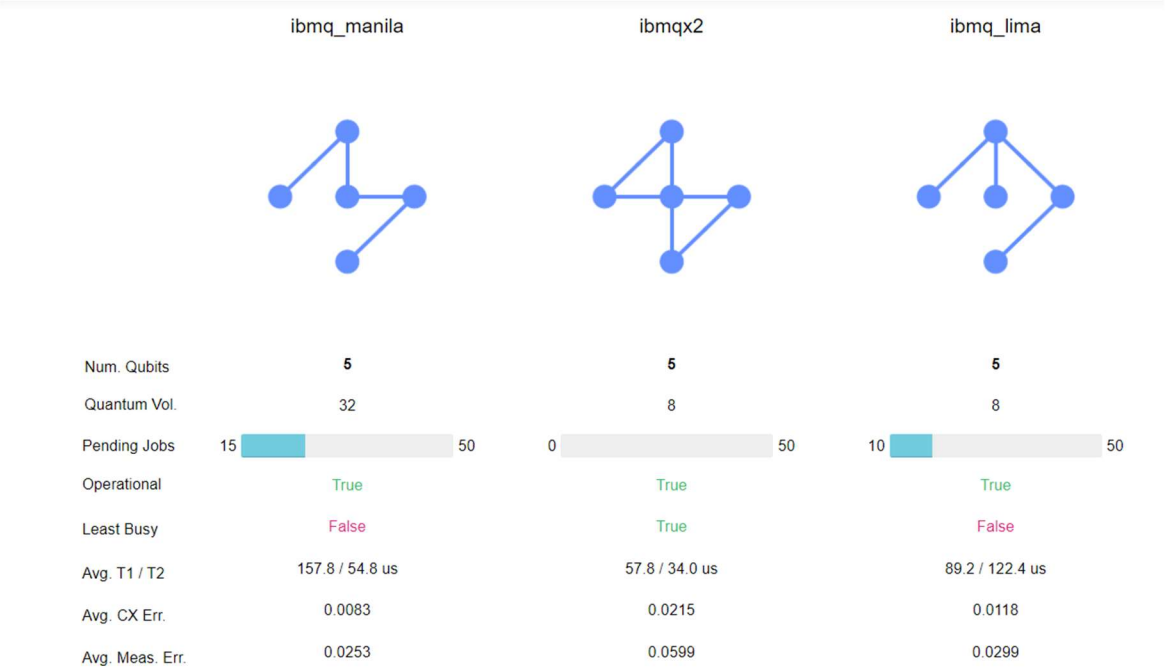
```
qc.cx(1,0)
```

d)
```
qc.h(inp[0])
for i in range(2):
    qc.cx(i-1,i)
qc.x([1,3])
```

42) Which of the following tools are used to get all the information on all the hardware available?

a) %qiskit_backend_overview
b) %qiskit.get_backend_overview
c) %backend_overview
d) %overview_backend

43) Using the information in the given figure below, choose the correct option. Which backend has the least average measurement error rate and has more pending jobs?

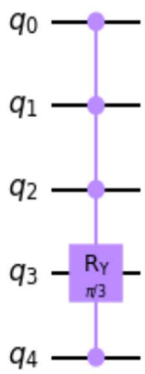


a) ibmqx2
b) ibmq_lima
c) ibmq_manila
d) None of the above

44) Which one of the following codes creates a pop of your jobs on your screen?

a) %qiskit_job_watcher
b) %qiskit.get_watcher_job
c) %qiskit.import_job_watcher
d) job_watcher

45) Which one of the given code fragments when executed gives the image shown below?



a) ry = RYGate(pi/3).control(4)
qc = QuantumCircuit(5)
qc.append(ry, [0,1,2,3,4]

b) ry = RYGate(pi/3).control(5)
qc = QuantumCircuit(5)
qc.append(ry, [0,1,2,4,3]

c) ry = RYGate(pi/3).control(4)
qc = QuantumCircuit(5)
qc.append(ry, [0,1,2,4,3]

d) ry = RYGate(pi/3).control(5)
qc = QuantumCircuit(5)
qc.append(ry, [0,1,2,3,4]

46) There are two qubits where an X gate is applied to the 2nd qubit and a H gate to the 1st qubit. What will be the output if they are measured?

a) 10 & 01
b) 00 & 11
c) 10 & 11
d) 01 & 00

47) Which one of the following codes given below when executed gives the same image?

q = QuantumRegister(2)
qc = QuantumCircuit(q)
#Insert your code here

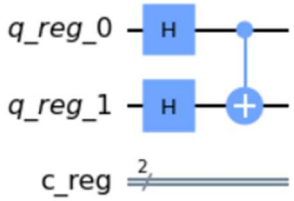
a) qc.swap(0,1)
b) qc.cz(0,1)
c) qc.cx(q[0],q[1])
d) qc.crz(0,0,1)

48) Which one of the following codes are used to monitor the job status?

a) job.status()
b) job.get_monitor()

c) job.get_status()
d) job.monitor()

49) Which one of the following codes when executed gives the image shown below as output?




```
q = QuantumRegister(2,'q_reg')
c = ClassicalRegister(2,'c_reg')
#Insert your code here
```

a)
```
qc = QuantumCircuit(q,c)
qc.h(q[0:2])
qc.cx(q[0], q[1])
```

b)
```
qc = QuantumCircuit(q,c)
qc.h(q[0:1])
qc.cx(q[1], q[0])
```

c)
```
qc = QuantumCircuit(q,c)
qc.h(q[2:0])
qc.cx(q[1], q[0])
```

d)
```
qc.h(q[0:1])
qc.cx(q[0], q[1])
```

50) Which of the following codes are used to call the real backend?
a) Aer.get_backend('ibmq_belem')
b) BasicAer.get_backend('ibmq_belem')
c) provider.get_backend('ibmq_belem')
d) All of the above

Answers

1	C	26	B,D,E
2	C	27	C
3	D	28	C
4	B	29	A
5	C	30	C
6	C	31	C
7	B	32	B
8	A	33	B
9	B	34	A
10	B	35	C
11	B	36	C
12	A	37	D

13	D	38	D
14	A	39	B
15	C	40	A
16	C	41	B
17	A	42	A
18	A	43	C
19	B,C	44	A
20	A	45	C
21	A	46	C
22	B	47	B
23	D,E	48	D
24	A	49	A
25	A,E	50	C