Quantum Computing – Notes Ver 1.2

Prepared By: Vijayananda Mohire

Sources: Various open courses, MOOC trainings and self study; no intention for any copyright infringements

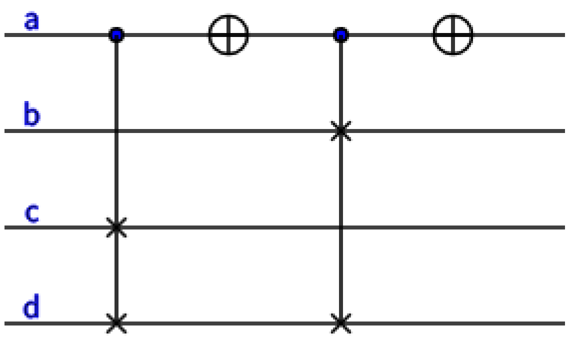
Question 1

Design a reversible circuit, using NOT, CNOT, Toffoli, and Fredkin gates, which acts on the four inputs a,b,c,d, to perform the operation $\text{swap}_{243}(a,b,c,d)$ which swaps b and d if $a=0$, and swaps c and d if $a=1$. Bit a should be left unchanged

Answer 1

High level function with the circuit

```
fredkin(a,c,d)
not(a)
fredkin(a,b,d)
not(a)
```



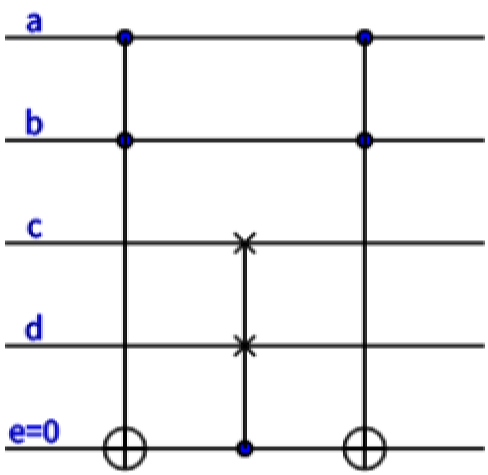
Question 2

Design a reversible circuit, using NOT, CNOT, Toffoli, and Fredkin gates, which acts on the four inputs a,b,c,d, to swap c and d only when both $a=1$ and $b=1$. You may use a fifth bit e, given as initialized to $e=0$, in your circuit; this bit must also end as $e=0$. C

Answer 2

High level function with the circuit

```
toffoli(a,b,e)
fredkin(e,c,d)
toffoli(a,b,e)
```



Question 3

Sample RandomNumber using Q#

Answer 3

```
open Microsoft.Quantum.Arrays;
open Microsoft.Quantum.Measurement;

operation SampleRandomNumber(nQubits : Int) : Result[] {
    // We prepare a register of qubits in a uniform
    // superposition state, such that when we measure,
    // all bitstrings occur with equal probability.
    use register = Qubit[nQubits] {
        // Set qubits in superposition.
        ApplyToEachA(H, register);

        // Measure all qubits and return.
        return ForEach(MResetZ, register);
    }
}
```

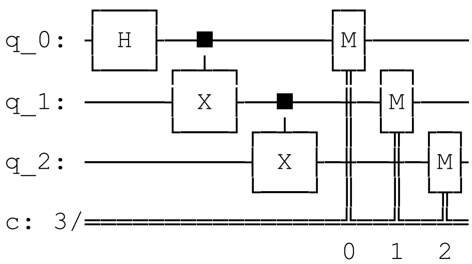
Question 4

Run a basic quantum circuit expressed using the [Qiskit library](#) to an IonQ target via the Azure Quantum service.

Answer 4

First, import the required packages for this sample:

```
from qiskit import QuantumCircuit
from qiskit.visualization import plot_histogram
from qiskit.tools.monitor import job_monitor
from azure.quantum.qiskit import AzureQuantumProvider

#Connect to backend Azure quantum service, using below function
from azure.quantum.qiskit import AzureQuantumProvider

provider = AzureQuantumProvider ( resource_id = " ", location = " " )

# Create a Quantum Circuit acting on the q register
circuit = QuantumCircuit(3, 3)
circuit.name = "Qiskit Sample - 3-qubit GHZ circuit"
circuit.h(0)
circuit.cx(0, 1)
circuit.cx(1, 2)
circuit.measure([0,1,2], [0, 1, 2])
# Print out the circuit
circuit.draw()
```



```
#Create a Backend object to connect to the IonQ Simulator back-end:
simulator_backend = provider.get_backend("ionq.simulator")

job = simulator_backend.run(circuit, shots=100)
job_id = job.id()
print("Job id", job_id)

#Create a job monitor object
job_monitor(job)

#To wait until the job is completed and return the results, run:
result = job.result()
```

```
qiskit.result.result.Result
```

```
print(result)

connect to real hardware (Quantum Processing Unit or QPU)
qpu_backend = provider.get_backend("ionq.qpu")

# Submit the circuit to run on Azure Quantum
qpu_job = qpu_backend.run(circuit, shots=1024)
job_id = qpu_job.id()
print("Job id", job_id)

# Monitor job progress and wait until complete:
job_monitor(qpu_job)

# Get the job results (this method also waits for the Job to complete):
result = qpu_job.result()
print(result)
counts = {format(n, "03b"): 0 for n in range(8)}
counts.update(result.get_counts(circuit))
print(counts)
plot_histogram(counts)
```

Question 5

Develop Google AI sample Cirq circuit

Answer 5

```
import cirq
qubits = [cirq.GridQubit(x, y) for x in range(3) for y in range(3)]
print(qubits[0])

# This is an Pauli X gate. It is an object instance.
x_gate = cirq.X
# Applying it to the qubit at location (0, 0) (defined above)
# turns it into an operation.
x_op = x_gate(qubits[0])
print(x_op)

cz = cirq.CZ(qubits[0], qubits[1])
x = cirq.X(qubits[2])
moment = cirq.Moment([x, cz])
x2 = cirq.X(qubits[2])
cz12 = cirq.CZ(qubits[1], qubits[2])
moment0 = cirq.Moment([cz01, x2])
```

```
moment1 = cirq.Moment([cz12])
circuit = cirq.Circuit((moment0, moment1))
print(circuit)
```

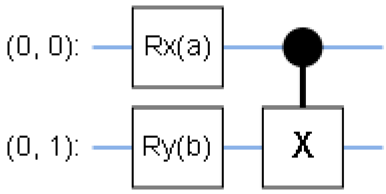
```
(0, 0): ───@───────
           │
(0, 1): ───@───@───
               │
(0, 2): ───X───@───
```

Question 6

Design a simple Tensorflow based quantum Colab sample

Answer 6

```
!pip install tensorflow==2.4.1
!pip install tensorflow-quantum

import tensorflow as tf
import tensorflow_quantum as tfq
import cirq
import sympy
import numpy as np
# visualization tools
%matplotlib inline
import matplotlib.pyplot as plt
from cirq.contrib.svg import SVGCircuit
a, b = sympy.symbols('a b')
# Create two qubits
q0, q1 = cirq.GridQubit.rect(1, 2)

# Create a circuit on these qubits using the parameters you created above.
circuit = cirq.Circuit(
    cirq.rx(a).on(q0),
    cirq.ry(b).on(q1), cirq.CNOT(control=q0, target=q1))
SVGCircuit(circuit)
```



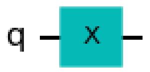

```
# Calculate a state vector with a=0.5 and b=-0.5.
resolver = cirq.ParamResolver({a: 0.5, b: -0.5})
output_state_vector = cirq.Simulator().simulate(circuit, resolver).final_state_vector
output_state_vector
```

Question 7

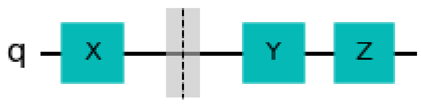
Design a simple qubit based quantum circuit using IBM Qiskit

Answer 7

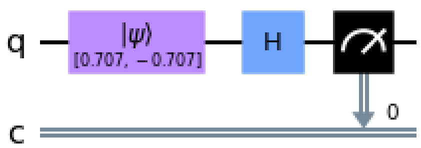
```
import numpy as np
# Importing standard Qiskit libraries
from qiskit import QuantumCircuit, transpile, Aer, IBMQ, assemble
from qiskit.tools.jupyter import *
from qiskit.visualization import *
from ibm_quantum_widgets import *
from math import pi, sqrt
# Loading your IBM Quantum account(s)
provider = IBMQ.load_account()
sim = Aer.get_backend('aer_simulator')

# Let's do an X-gate on a |0> qubit
qc = QuantumCircuit(1)
qc.x(0)
qc.draw()
```




```
qc.y(0) # Do Y-gate on qubit 0
qc.z(0) # Do Z-gate on qubit 0
qc.draw()
```




```
# Create the X-measurement function:
def x_measurement(qc, qubit, cbit):
"""Measure 'qubit' in the X-basis, and store the result in 'cbit'"""
qc.h(qubit)
qc.measure(qubit, cbit)
return qc
initial_state = [1/sqrt(2), -1/sqrt(2)]
# Initialize our qubit and measure it
qc = QuantumCircuit(1,1)
qc.initialize(initial_state, 0)
x_measurement(qc, 0, 0) # measure qubit 0 to classical bit 0
qc.draw()
```



Question 8

How to find if matrix is Unitary

Answer 8

Consider a 2*2 Matrix A with different values. We take 2 examples as shown below to prove how these are valid or not for quantum representation

A = $\begin{matrix} 0 & 1 \\ i & 0 \end{matrix}$ and A^T = $\begin{matrix} 0 & i \\ 1 & 0 \end{matrix}$

Next, A^*A^T = $\begin{matrix} i & 0 \\ 0 & i \end{matrix}$ = i * $\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$ which is an Identity matrix I

So this matrix is **Unitary** and valid for quantum representations

Next example,
A = $\begin{matrix} 1 & -1 \\ 0 & 1 \end{matrix}$ and A^T = $\begin{matrix} 1 & 0 \\ -1 & 1 \end{matrix}$

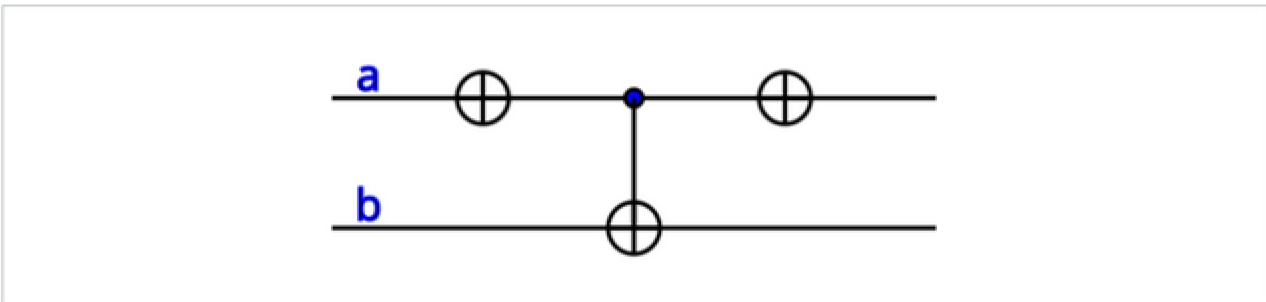
Next, A^*A^T = $\begin{matrix} 2 & 0 \\ -1 & 0 \end{matrix}$ = which is NOT an Identity matrix, as 2 is not correct

So this matrix is **NOT unitary** and NOT valid for quantum representations

Question 9

Generate the Unitary matrix for the given quantum circuit

Consider the following quantum circuit *C*, composed of two NOT gates and one CNOT gate:



Answer 9

First let me get the matrices for NOT and CNOT gates

NOT = $\begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix}$ and for CNOT $\begin{pmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\end{pmatrix}$

Gate Matrices have to be multiplied. However, when matrix is generated for single qubit ,tensor product with identity is required.

So getting the I for the NOT gates

$\begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix}$ tensor product $\begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix}$ = $\left(\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\right)$ this is the Identity I

Now multiply these as per circuit order

I * CNOT Matrix *I

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix}$$

The multiplication can be made easier using online tool like

https://www.dcode.fr/matrix-multiplication

This is based on theory, however this needs to be done using simulator like Qiskit based Composer and get the Unitary matrix

Question 10: Derive Pauli's X gate

Answer 10: There are 3 Pauli's gates namely X, Y and Z that represent the various gate operations on the Bloch sphere.

Pauli's X gate offer a NOT type of operation and is represented by bra-ket and matrix notations. Below is an example of deriving the X gate

Please note bra is represented by < 0 | and ket by |0 >. Arranging the matrices in proper shape is the key in getting the proper results. There is also a conjugate transpose required, meaning the cols matrix is transformed to row matrix and these are then multiplied

I have used a different method to represent the state vector rows and columns; however this is not the best one. You can use KET based COLS first and BRA based ROWS, and then do the operation. Pauli X is a NOT gate, so the 0->1 and 1>0 are reflected in the matrices. Please get these things clear first

Pauli's X = NOT = $|0\rangle\langle 1| + |1\rangle\langle 0| = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$ ⇒ value 1
$(0 \to 1, 1 \to 0)$

Representing Bra-ket in matrix form

$$\underbrace{|0\rangle\langle 1|}_{I} + \underbrace{|1\rangle\langle 0|}_{II} = \underbrace{\begin{bmatrix}1\\ 0\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix}}_{I} + \underbrace{\begin{bmatrix}0\\ 1\end{bmatrix}\begin{bmatrix}1 & 0\end{bmatrix}}_{II}$$

ket (cols) Bra (rows) (~~cols~~)

These rows are conjugate transpose.

Now, multiply the 2 matrices of I

$$\begin{bmatrix}1\\ 0\end{bmatrix}\begin{bmatrix}0 & 1\end{bmatrix} = \begin{bmatrix}1\times 0 & 1\times 1\\ 0\times 0 & 0\times 1\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}$$ ⇒ computed ~~Required~~ Value of I (a)

||ly, multiply the 2 matrices of II

$$\begin{bmatrix}0\\ 1\end{bmatrix}\begin{bmatrix}1 & 0\end{bmatrix} = \begin{bmatrix}0\times 1 & 0\times 0\\ 1\times 1 & 1\times 0\end{bmatrix} = \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix}$$ ⇒ computed val of II. (b)

Now adding I & II (I + II)

$$\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix} + \begin{bmatrix}0 & 0\\ 1 & 0\end{bmatrix} = \begin{bmatrix}0+0 & 1+0\\ 0+1 & 0+0\end{bmatrix} = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$$ ⇒ This is the value 1

Hence the above method to solve the Paul's X is correct.

Question 11: Derive Pauli's Y gate

Answer 11: In a similar way the Pauli's X is derived, Pauli's Y is derived

Pauli Y :- $\sigma_y = \underbrace{i|1\rangle\langle 0|}_{I} - \underbrace{i|0\rangle\langle 1|}_{II}$

Should result in $\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$

(I) → matrix form.

$$i|1\rangle\langle 0| = + i\begin{bmatrix} 0 \\ 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \end{bmatrix} = i\begin{bmatrix} 0\times 1 & 0\times 0 \\ 1\times 1 & 1\times 0 \end{bmatrix} =$$

$$i\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \rightarrow I^{st} \text{ term}$$

(II) → matrix form.

$$i|0\rangle\langle 1| = i\begin{bmatrix} 1 \\ 0 \end{bmatrix}\begin{bmatrix} 0 & 1 \end{bmatrix} = i\begin{bmatrix} 1\times 0 & 1\times 1 \\ 0\times 0 & 0\times 1 \end{bmatrix}$$

$$= i\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

(I) minus (II)

$$i\begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} - i\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \equiv \begin{bmatrix} 0 & 0 \\ i & 0 \end{bmatrix} - \begin{bmatrix} 0 & i \\ 0 & 0 \end{bmatrix}$$

Adding

$$\equiv \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \checkmark \Rightarrow \text{Desired result.}$$

Question 12: Derive Pauli's Z gate, **Answer 12**

$$X = \sigma_x = NOT = |0\rangle\langle 1| + |1\rangle\langle 0| = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$Z = \sigma_z = \text{Signflip} = |0\rangle\langle 0| - |1\rangle\langle 1| = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$Y = \sigma_y = -i|0\rangle\langle 1| + i|1\rangle\langle 0| = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

uv^T outer products (Tensor product) $|0\rangle\langle 1|$ → kets first (col) bra ~~rows~~ next (row)

Inner products $\langle 0|1\rangle$ → bras ~~rows~~ first (row). kets next (col).

u^Tv (Dot product) o/ps scalar.

Kronecker product (pair of matrices (i/p), o/p → block matrix).

— Similar to outer product.

Z is an ~~orthog~~ diagonal representation (orthonormal decomposition), so using bracket is not correct ans.

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = |0\rangle\langle 0| - |1\rangle\langle 1|$$

Question 13: Show an example of inner product

Answer 13: Inner product of 2 matrices is the dot product and results in a scalar.

Inner & outer products: (for Column vectors ONLY)

Inner

bra-ket

$u = \begin{bmatrix}-1\\3\end{bmatrix}$ $v = \begin{bmatrix}2\\5\end{bmatrix}$ $u^T v = \begin{bmatrix}-1 & 3\end{bmatrix}\begin{bmatrix}2\\5\end{bmatrix} = -1\cdot 2 + 3\cdot 5 = -2+15 = \underline{\underline{13}}$

Scalar

outer $uv^T = \begin{bmatrix}-1\\3\end{bmatrix}\begin{bmatrix}2 & 5\end{bmatrix} = \begin{bmatrix}-2 & -5\\6 & 15\end{bmatrix}$ → vector matrix. (Tensor)

ket-bra

Inner is bra-ket, outer is ket-bra

Ex. pauli X → Inner & outer products

Inner

say, $u = \begin{bmatrix}-1\\3\end{bmatrix}$ and

$X = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$

$u^T X = \begin{bmatrix}-1 & 3\end{bmatrix}\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} = \begin{bmatrix}0 & -1\\3 & 0\end{bmatrix}$

Inner products are Scalar

$= 0+3-1+0 = \underline{2}$ is Inner product with pauli X.

Question 14: Show an example of outer product

Answer 14: Outer product of 2 matrices is the tensor product and results in a vector matrix.

say, $u = \begin{bmatrix}-1\\3\end{bmatrix}$

Inner

$X = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$

$u^T X = \begin{bmatrix}-1 & 3\end{bmatrix}\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} = \begin{bmatrix}0 & -1\\3 & 0\end{bmatrix}$

Inner products are Scalar

$= 0+3-1+0 = \underline{\underline{2}}$ is Inner product with pauli X.

outer

$u = \begin{bmatrix}-1\\3\end{bmatrix}$

$x = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$ → transpose

$ux^T = \begin{bmatrix}-1\\3\end{bmatrix}\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} = \begin{bmatrix}0 & -1\\3 & 0\end{bmatrix}$ = tensor product

outer products are tensor product

Question 15: Show an example of outer product using Pauli X & Y with an example of Trace

Answer 15: **U**sing Pauli's X & Y matrices

Tensor product of Pauli x & y is (outer product).

$$X \otimes Y = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} = \begin{bmatrix} 0 \cdot \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} & 1 \cdot \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \\ 1 \cdot \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} & 0 \cdot \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ i & 0 & 0 & 0 \end{bmatrix} \checkmark$$

⇒ Kronecker is also same steps.

Trace of a matrix is $tr(A^T)$ → sum of the main diagonal entries

Ex:

$$A = \begin{bmatrix} -1 & 2 & 4 \\ 3 & 7 & 0 \\ 5 & 8 & -6 \end{bmatrix} \quad A^T = \begin{bmatrix} -1 & 3 & 5 \\ 2 & 7 & 8 \\ 4 & 0 & -6 \end{bmatrix} \quad tr(A^T) = -1 + 7 - 6 = 0$$

Rows → cols for AT

Question 16: Show how Bell State is derived

Answer 16: Bell state preparation uses 3 steps:

1. State initialization
2. Use Hadamard and Identity gate for superposition and getting the Kronecker matrix
3. Use a CNOT to multiply with the Kronecker matrix

Details in the following notes below

Bell state:

$$CNOT\ (H \otimes I)\ (|0\rangle \otimes |0\rangle)$$

step3 ← step2 ← step1

Step1 :- $|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ → first qbit & second qbit initi to zero

→ state matrix.

Step 2:

$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ & $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Kronecker / outer product = $H \otimes I$ =

$$H \otimes I = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & 1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 1 \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & -1 \begin{bmatrix} 1 & 0 \\ 0 & +1 \end{bmatrix} \end{bmatrix}$$

$$= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

~~Step3~~ Now multiply step 1 & 2.

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1+0+0+0 \\ 0+0+0+0 \\ 1+0-0+0 \\ 0+0+0-0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

This result needs to be applied CNOT. in step3.

Step3.

CNOT matrix = $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

so,

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1+0+0+0 \\ 0+0+0+0 \\ 0+0+0+0 \\ 0+0+1+0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Desired Bell state for first qubit state

Question 17: State the types of quantum states

Answer 17: Quantum qubit can have 6 possible states, 2 each for the X, Y and Z directions of the Bloch sphere

ket state rep (General rep)

for first qubit state $|00\rangle$

$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \rightarrow$ Superposition state

4 states

$|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \rightarrow$ plus state. $= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

$|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \Rightarrow$ minus state

$|i\rangle = \frac{1}{\sqrt{2}}|0\rangle + i\frac{1}{\sqrt{2}}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \rightarrow$ i state.

$|-i\rangle = \frac{1}{\sqrt{2}}|0\rangle - i\frac{1}{\sqrt{2}}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \Rightarrow$ -i state

These are total 6 Quantum states for a given qubit. $\underbrace{|0\rangle, |1\rangle}_{\text{For z axis}}, \underbrace{|+\rangle, |-\rangle}_{\text{for x axis}}, \underbrace{|i\rangle \text{ and } |-i\rangle}_{\text{For y axis.}}$

Another way to represent these are shown below, |0>, |1>,| +>, | ->,| I > and | –I >

$$\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \alpha\begin{pmatrix}1\\0\end{pmatrix} + \beta\begin{pmatrix}0\\1\end{pmatrix}$$
$$= \alpha|0\rangle + \beta|1\rangle$$
$$= |\Psi\rangle \text{ "ket"}$$

$$\langle\Psi| \text{ "bra"}$$
$$\langle\Psi| = (|\Psi\rangle)^{\dagger} = (|\Psi\rangle)^{*T}$$
$$\langle\Psi| = \begin{pmatrix}\alpha^*\\ \beta^*\end{pmatrix}^{T}$$
$$= (\alpha^* \ \ \beta^*)$$
$$= \alpha^*(1 \ \ 0) + \beta^*(0 \ \ 1)$$
$$= \alpha^*\langle 0| + \beta^*\langle 1|$$

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad \text{"superposition state"}$$
$$|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$
$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{"plus state"}$$
$$|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$
$$\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad \text{"minus state"}$$
$$|i\rangle = \frac{1}{\sqrt{2}}|0\rangle + i\frac{1}{\sqrt{2}}|1\rangle$$
$$\frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \quad \text{"i state"}$$
$$|-i\rangle = \frac{1}{\sqrt{2}}|0\rangle - i\frac{1}{\sqrt{2}}|1\rangle$$
$$\frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \quad \text{"}-i\text{ state"}$$

Image source: https://andisama.medium.com/qubit-an-intuition-1-first-baby-steps-in-exploring-the-quantum-world-16f693e456d8

Question 18: Define the notations for the different types of quantum states like plus, minus etc

Answer 18: Quantum qubit state notations are mainly represented in matrix and bra-ket forms with transformation from one notation to another as required to solve a problem .Below are matrix notations for 0,1, + and – states. These can be re-written from matrix to state, like col matrix [1 0] can be written as ket notation | 0> as per the need of the problem to be solved

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} \Rightarrow |+\rangle$$

$$\frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ -1 \end{bmatrix} = |-\rangle$$

Question 19: Apply an H gate on the |+> and show the results

Answer 19: First we get the matrix notation for H and |+> states, then we multiply them, details shown below

Apply a H gate on the $|+\rangle$ state.

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H|+\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$\Rightarrow$ see prev. pg

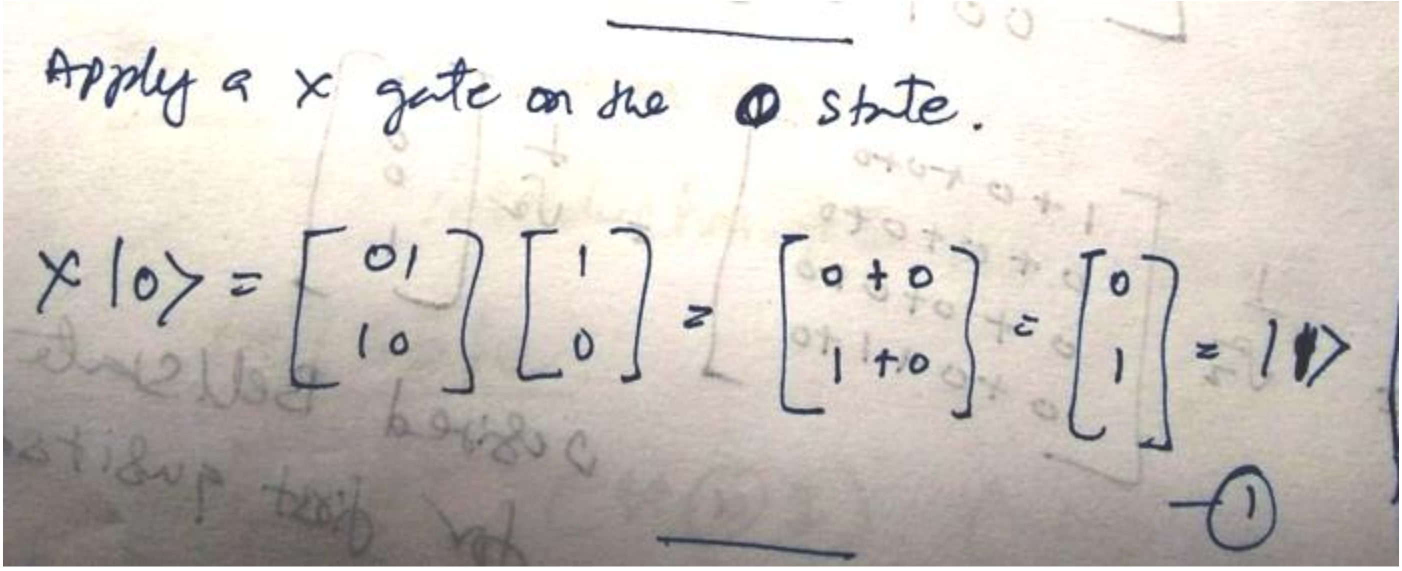
$$H|+\rangle = \frac{1}{2}\begin{bmatrix} 2 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Rows of first matrix multiplied by cols of sec. matrix.

$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ state matrix represents $|0\rangle$ state ket.

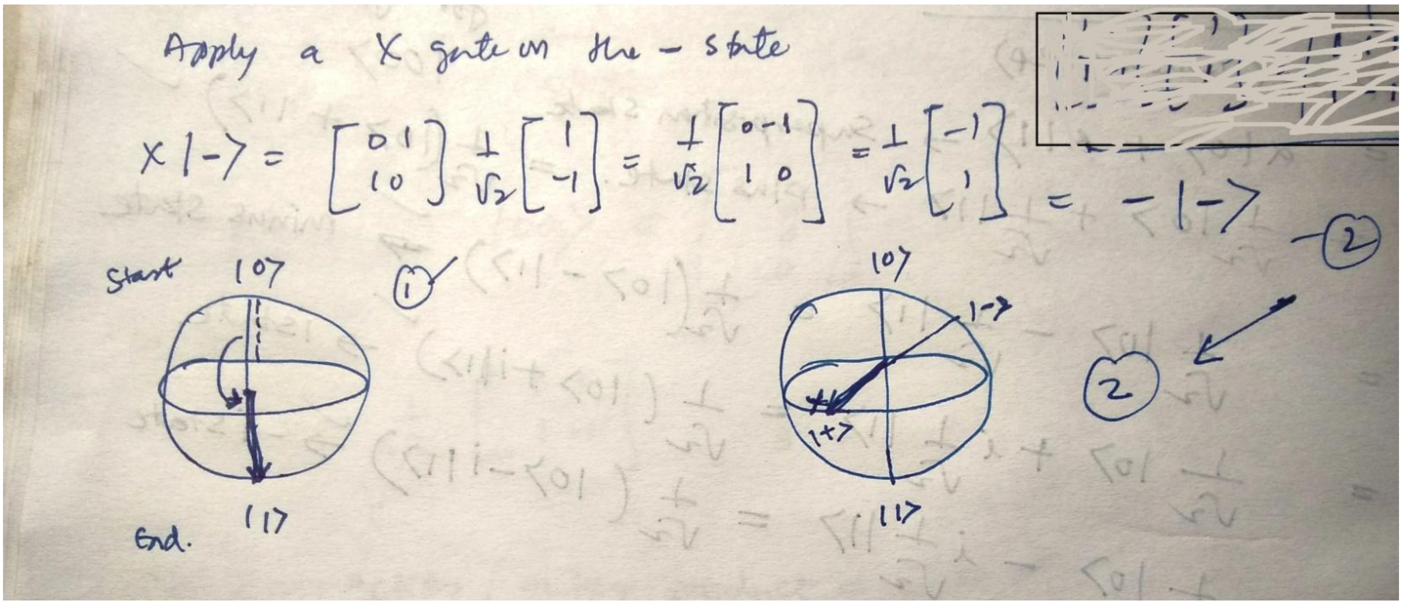
Question 20: Apply an X gate on the |0> and show the results

Answer 20: First we get the matrix notation for X and |0> states, then we multiply them, as shown below



Question 21: Apply an X gate on the |-> and show the results

Answer 21: First we get the matrix notation for X and |-> states, then we multiply them, details shown below ,results show on Bloch sphere for Question 19 and 20



Question 22: Test the below matrices for the validity of being the bitflip X gate

Answer 22: First we get the matrix notation of the X gate and test it against each given matrix that should result in the NOT operation

Which of these are valid bit flip X gate

A) $\begin{bmatrix}0&0\\0&1\end{bmatrix}$ B) $\begin{bmatrix}0&0\\0&1\end{bmatrix}$ C) $\begin{bmatrix}1&0\\0&-1\end{bmatrix}$ D) $\begin{bmatrix}0&1\\1&0\end{bmatrix}$ ✗

Test:

A) $\begin{bmatrix}0&0\\0&1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0+0\\0+0\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix} \neq \begin{bmatrix}0\\1\end{bmatrix}$ → which is required to flip the $\begin{bmatrix}1\\0\end{bmatrix}$ state

B) $\begin{bmatrix}0&0\\0&1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0+0\\0+0\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}$ ↗ same issue ✗

C) $\begin{bmatrix}1&0\\0&-1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}1+0\\0-0\end{bmatrix} = \begin{bmatrix}1\\0\end{bmatrix}$ → Does not flip ✗

D) $\begin{bmatrix}0&1\\1&0\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \begin{bmatrix}0+0\\1+0\end{bmatrix} = \begin{bmatrix}0\\1\end{bmatrix}$ ⇒ ✓ as $\begin{bmatrix}1\\0\end{bmatrix}$ is now $\begin{bmatrix}0\\1\end{bmatrix}$

Question 23: Given H acting on |0> produces |+> & H|1> = |->, which is the correct H operator

Answer 23: First we get the matrix for H and test each given matrix that produces the required results

Given H acting on |0> produces |+> & H|1> = |->, which is correct H operator.

A) $\frac{1}{\sqrt{2}}\begin{pmatrix}0&1\\1&0\end{pmatrix}$ B) $\frac{1}{\sqrt{2}}\begin{pmatrix}1&-1\\-1&1\end{pmatrix}$ C) $\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}$ D) $\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\-1&-1\end{pmatrix}$

Test:

A) $\frac{1}{\sqrt{2}}\begin{bmatrix}0&1\\1&0\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0+0\\1+0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\end{bmatrix}$ ⇒ this is NOT |+> ⇒ $\frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$ ✗

B) $\frac{1}{\sqrt{2}}\begin{bmatrix}1&-1\\-1&1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1-0\\-1\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}$ ⇒ same ✗

C) $\frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1+0\\1-0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$ ⇒ correct as this is |+>

✓ chk for |->

$\frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0+1\\0-1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}$ ⇒ correct as this is |->

Question 24: Express |+> state in the Z – basis (Hadamard)

Answer 24:

is needed for ... basis

State decomposition:

$$|\phi\rangle = \sum_k \langle\psi_k|\phi\rangle\,|\psi_k\rangle$$ → Express one state notation in another basis

Example: $|\phi\rangle = |0\rangle$, $|\psi_0\rangle = |+\rangle$, $|\psi_1\rangle = |-\rangle$

SANDWICH

So,

$$\sum_k \langle\psi_k|\phi\rangle|\psi_k\rangle = \underbrace{\langle+|0\rangle|+\rangle}_{I} + \underbrace{\langle-|0\rangle|-\rangle}_{II}$$

Replace +, 0,

$$I \quad \langle+|0\rangle = \frac{1}{\sqrt{2}}[1,1]\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}$$

Since $\langle+|$ is bra → we have row, since $|0\rangle$ is ket we have col

Illy

$$II \quad \langle-|0\rangle = \frac{1}{\sqrt{2}}[1,-1]\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}$$

So, $$|0\rangle = I + II = \frac{1}{\sqrt{2}}|+\rangle + \frac{1}{\sqrt{2}}|-\rangle = \frac{1}{\sqrt{2}}(|+\rangle + |-\rangle)$$

Thus the Hadamard basis is expressed in terms of x basis.

Question 25: Using Matrix and related gates derive Bell states

Answer 25: Please refer images below

Try matrix for BELL STATES

$$H \otimes I = \frac{1}{\sqrt{2}}\begin{bmatrix}1&1\\1&-1\end{bmatrix}\begin{bmatrix}1&0\\0&1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{bmatrix}$$

For first qubit: use calc.

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{bmatrix}\begin{bmatrix}1\\0\\0\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\1\\0\end{bmatrix}$$ ← First qubit

Apply CNOT to get final result.

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}1\\0\\1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\0\\1\end{bmatrix} = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

→ states bits 00 & 11 are ON.

Bell State 1

For second qubit:

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{bmatrix}\begin{bmatrix}0\\1\\0\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\\0\\1\end{bmatrix}$$

Apply CNOT

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}0\\1\\0\\1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\\1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}(|01\rangle+|10\rangle)$$

→ states bits 01 & 10 are ON

✓ II

For third qubit:

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{bmatrix}\begin{bmatrix}0\\0\\1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\-1\\0\end{bmatrix}$$

→ states the bits 00 & -11 are ON

Apply CNOT

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}1\\0\\-1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\0\\-1\end{bmatrix} = \frac{|00\rangle-|11\rangle}{\sqrt{2}}$$

✓ III

For forth qubit:

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{bmatrix}\begin{bmatrix}0\\0\\0\\1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\\0\\-1\end{bmatrix}$$

Apply CNOT

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}0\\1\\0\\-1\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}0\\1\\-1\\0\end{bmatrix} = \frac{|01\rangle-|10\rangle}{\sqrt{2}}$$

→ states 01 & -10 on

IV

Question 26: Show the Eigen vectors and Eigen values for Paulis XYZ

Answer 26: Eigen values in each case are + and –. Eigen vectors are shown below

Eigen vect & eigen vals of Pauli's : eigen

eigenstates.

$$\psi_{x+} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix} = \frac{1}{\sqrt{2}}(|1\rangle + |1\rangle)$$

$$\psi_{x-} = \frac{1}{\sqrt{2}}\begin{bmatrix}-1\\1\end{bmatrix} = \frac{1}{\sqrt{2}}(-|1\rangle + |1\rangle)$$

$$\psi_{y+} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\i\end{bmatrix} = \frac{1}{\sqrt{2}}(|1\rangle + |i\rangle)$$

$$\psi_{y-} = \frac{1}{\sqrt{2}}\begin{bmatrix}i\\1\end{bmatrix} = \frac{1}{\sqrt{2}}(|i\rangle + |1\rangle)$$

$$\psi_{z+} = \begin{bmatrix}1\\0\end{bmatrix} = |1\rangle + |0\rangle \checkmark$$

$$\psi_{z-} = \begin{bmatrix}0\\1\end{bmatrix} = |0\rangle + |1\rangle \checkmark$$

Question 27: Please test if these states are separable?

Answer 27: Please refer image below

Are the states separable ? in terms of entanglement.

$$\frac{|00\rangle + |01\rangle}{\sqrt{2}} = |0\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle \quad \text{yes} \checkmark$$

$$\frac{|{+}{+}\rangle + |{-}{-}\rangle}{\sqrt{2}} = \quad \text{No common, so not seperable.} \times$$

Question 28: Show the probability of finding a qubit in a given state

Answer 28: Please refer image below

Probability of finding qubit in state

$$R_y(\theta) = \exp\left(-i\frac{\theta}{2}Y\right) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\theta/2 & \cos\theta/2 \end{pmatrix} \qquad = -i\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

I/p state $|0\rangle$

prob. of getting $|0\rangle$ as measurement is:

square of the abs val of first row + first col.

$\cos^2\frac{\theta}{2}$ replace θ with $3\pi/4$

$$\cos^2\left(\frac{3\pi}{4\times 2}\right) = \cos^2\left(\frac{3\pi}{8}\right) = \cancel{\cos^2(0.38268)} = 0.146$$

Question 29: Show unitary rotation matrices around Pauli XYZ

Answer 29: Please refer image

einstienrelatively easy.com

Rotation:

$$R_x(\theta) = e^{-i\frac{\theta}{2}X} = \cos\left(\frac{\theta}{2}\right)I - i\sin\left(\frac{\theta}{2}\right)\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\theta/2 \end{bmatrix}$$

$$R_y(\theta) = e^{-i\frac{\theta}{2}Y} = \cos\left(\frac{\theta}{2}\right)I - i\sin\left(\frac{\theta}{2}\right)\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} = \begin{bmatrix} \cos\theta/2 & -\sin\theta/2 \\ \sin\theta/2 & \cos\theta/2 \end{bmatrix}$$

$$R_z(\theta) = e^{-i\frac{\theta}{2}Z} = \cos\left(\frac{\theta}{2}\right)I - i\sin\left(\frac{\theta}{2}\right)\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} \cos\frac{\theta}{2} - i\sin\frac{\theta}{2} & 0 \\ 0 & \cos\frac{\theta}{2} + i\sin\frac{\theta}{2} \end{bmatrix}$$

All pauli matrix one notation

$$\sigma_j = \begin{pmatrix} \delta_{j3} & \delta_{j1} - i\delta_{j2} \\ \delta_{j1} + i\delta_{j2} & -\delta_{j3} \end{pmatrix}$$

Pauli vector provides mapping ~~of~~ from vector basis to Pauli matrix basis

$$\varphi = \begin{pmatrix} a_3 & a_1 - ia_2 \\ a_1 + ia_2 & -a_3 \end{pmatrix} \qquad A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

$$= a_1\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + a_2 i\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} + a_3\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

References:

1. MIT OpenCourseWare , https://ocw.mit.edu/
2. IBM Quantum Lab, https://quantum-computing.ibm.com/lab
3. Azure Quantum, https://azure.microsoft.com/en-in/services/quantum/
4. QuTech Academy, https://www.qutube.nl/
5. Andi Sama Blog, https://andisama.medium.com/qubit-an-intuition-1-first-baby-steps-in-exploring-the-quantum-world-16f693e456d8
6. Einstein Relatively Easy, https://einsteinrelativelyeasy.com/

Disclaimer: I have no intention for any copyright infringement, nor I promise that the results are true and right. Please use your caution to self-check the results against the quantum postulates. I am reachable at vijaymohire@gmail.com for any clarifications