Quantum Computing Full

March 7, 2024

1 Quantum Computing

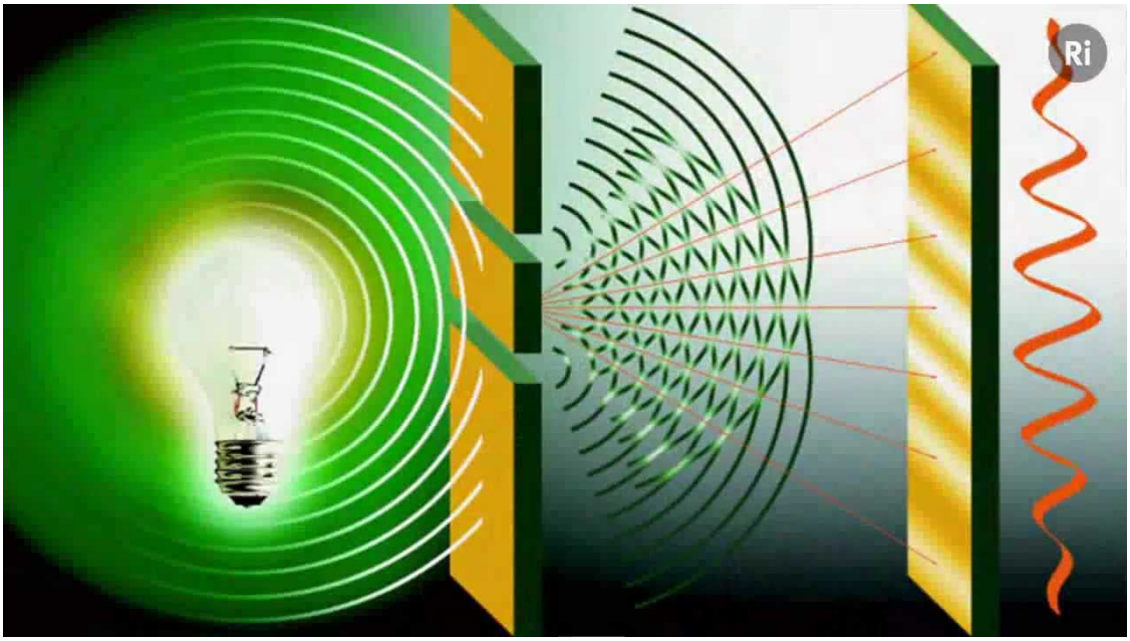
2 Wave Particle Duality :

In the realm of quantum computing, understanding the dual nature of matter is fundamental to comprehending the behavior of particles at the quantum level, which forms the basis of qubits, the building blocks of quantum computers. Here are some key points:

1. **Wave-Particle Duality:** Particles at the quantum level, such as electrons and photons, exhibit both wave-like and particle-like behaviors. This duality challenges classical physics, where particles were considered distinct from waves. In quantum mechanics, particles can display wave-like characteristics, described by wave functions.
2. **Quantum Superposition:** Particles in a quantum state can exist in multiple states simultaneously, a concept known as superposition. This property is crucial in quantum computing because qubits can represent both 0 and 1 at the same time, enabling parallel computation.
3. **Interference:** When waves interact, they can interfere constructively (increasing amplitude) or destructively (canceling each other out). In quantum computing, qubits exploit interference patterns to enhance or minimize specific outcomes, crucial for quantum algorithms like Shor's algorithm or Grover's algorithm.
4. **Measurement and Collapse of the Wavefunction:** When a quantum system is measured, it collapses into one of its possible states. Before measurement, the system exists in a superposition of states. This property allows quantum computers to perform multiple calculations simultaneously but requires careful handling to preserve the quantum state.
5. **Quantum Entanglement:** Particles can become entangled, where the state of one particle becomes correlated with the state of another, regardless of the distance between them. Entanglement is a powerful resource in quantum computing, enabling the creation of highly interconnected qubits for faster computations.
6. **Decoherence:** Quantum systems are fragile and can easily lose their quantum properties due to interactions with the environment. This loss of coherence, known as decoherence, poses a significant challenge in maintaining the stability of qubits and sustaining their quantum state.

Understanding the dual nature of matter provides insights into the behavior of particles at the quantum level, forming the theoretical basis for quantum computing. Harnessing these principles effectively is crucial for building and operating robust and efficient quantum computers.

3 Young's Double-slit Experiment :



The double-slit experiment is a cornerstone in understanding the dual nature of matter, revealing the wave-particle duality in quantum mechanics.

Double-Slit Experiment:

In this experiment:

- A light source (or any particle source, like electrons) shines particles toward a barrier with two slits.
- Behind the barrier, there's a screen to detect where the particles land after passing through the slits.

Observations:

1. **Particle-Like Behavior:** When one slit is open, particles land on the screen directly behind that slit, behaving as if they were tiny bullets going through a single slit.
2. **Wave-Like Behavior:** When both slits are open, an interference pattern emerges on the screen, similar to what waves produce when they pass through two slits and interfere with each other. This pattern indicates that particles are behaving like waves, creating areas of reinforcement (constructive interference) and cancellation (destructive interference).

Significance and Implications:

The double-slit experiment shook up classical physics because it demonstrated that particles, like electrons or photons, exhibit behavior characteristic of waves. This duality challenges the traditional view of particles as tiny, solid objects and instead suggests that they also have wave-like properties.

Thought behind the Discovery of Dual Nature:

This experiment led to the concept of "wave-particle duality." The thought was that particles (like electrons) could exhibit wave-like behaviors—such as interference patterns—suggesting they weren't just discrete particles but had some wave-like characteristics as well.

The discovery of this duality marked a fundamental shift in our understanding of the nature of matter. It laid the groundwork for quantum mechanics, showing that at the smallest scales, particles don't adhere to classical physics' strict division between particles and waves. Instead, they exhibit both behaviors simultaneously, which is key to comprehending the behavior of particles at the quantum level.

4 Introduction :

4.1 Why Quantum Computing?

Modern computing is based on `classical physics` and `Mathematical logic` laws. Even though the working of electronic components works on quantum mechanical principles, the logic they follow is `classical`. Traditional computer software is designed for **`serial computation`**.

`Serial computation` essentially means that when we write an algorithm, the logic flow takes place from one point to another in terms of time. It means that a particular process must be completed before another process is taken up.

The concept of a **`parallel computer`** is also within our traditional computing platform, but the problem can be broken up into independent logic which could be executed at the same time. So in other words, a traditional parallel computer must have an `n` number of processors which could take up the job at the same time and then we should integrate with the results.

The Idea of Quantum Computer was first given by `Richard Feynmann` in a paper which he published in 1982. That paper was titled **`"Simulation of Physics in a Computer"`**.

The question that he asked is **Can one simulate physics using a computer?**

Now there are many problems with it, the first problem is `how do you simulate time?` - As we know time is a continuous variable. The problem with stimulating time is to discretize it. We have techniques for discretizing the time by solving **Differential Equations**.

But the bigger problem is that, in quantum mechanics, the measurements will give **probabilistic** results.

- As we know a state in quantum mechanics is a `linear combination` of certain states and when a measurement is made, let's consider about linear combination of some `basis` states when we measure any physical property of the system, one of the possible the values of that physical property is realized with certain probability.

But then the problem is to simulate a `quantum process` using a quantum computer *Can we use the quantum principle to our advantage to build a computer?*

4.2 Peter Shor's Approach :

In 1994 he came up with an algorithm which showed that a very old problem that we have in computational science can be effectively solved with a quantum computer. The problem is connected with how to factorize a large composite number.

Now this has been known to be a very difficult problem or as computer scientists call it `hard problem` in computer science.

- The reason why it is a hard problem is that there are no effective algorithms known, which can compute the factors of a large composite number which in computer science language is referred to as `polynomial time`.

If it could be done in polynomial time then of course it would be called an `easy problem`.

Even today we depend on this difficulty in factorizing a large composite number to have encryption of data and in fact ***`RSA algorithm`*** which provides the data encryption depends on the relative hardness of the factorization problem and w.r.t the multiplication which is relatively easy.

- And if one can break this RSA Code which at least theoretically is possible today thanks to Shor's algorithm, then it would mean a substantial advancement in both cryptography and in computer science.

So what **Peter Shor** showed is that using the principles of quantum mechanics we can *factorize a large composite number.*

4.3 Inherent Parallelism :

In a Quantum Computer, the same processor can perform operations on multiple inputs simultaneously and the state of a register can exist in a simultaneous superposition of different Quantum states.

The reason why quantum computation is different from classical computation is the fact that while a classical register can at a given time have or be in one state i.e., let us consider a simple classical bit and I have a one-bit register.

1. `States and Superposition`:
    - In classical computation, a **bit** in a register can exist in one of two states: `0` or `1`. For example, a **2-bit** register can represent one of the four classical states: `00`, `01`, `10`, or `11`.
    - In quantum computation, quantum bits or qubits can exist in a state of superposition. This means a qubit can represent both 0 and 1 simultaneously, in varying proportions. For instance, a 2-qubit quantum register can be in a linear combination of the classical states (00, 01, 10, 11) at the same time.
2. `Superposition in Computation`:
    - When performing computations on classical bits, you can only process `one state at a time`. If you have multiple inputs, you'd need `multiple processors or sequential operations` to compute the function for each input individually.
    - Quantum computation allows for computation on **superpositions**. This means when a function is applied to a quantum register in superposition, the computation is carried out on all the possible inputs simultaneously due to the nature of quantum parallelism. This inherent parallelism enables computations on multiple inputs at once, without the need for separate processors.
3. `Parallelism`:

- In classical computing, parallelism is achieved by using `multiple processors`, where different processors handle different computations **concurrently**. This external parallelism involves dividing tasks among processors to speed up computation.
- Quantum parallelism, on the other hand, `is inherent to the nature of qubits and superposition`. Quantum algorithms take advantage of this intrinsic property, allowing for computations on multiple states simultaneously without requiring additional hardware.
- And the parallelism that we are talking about for a quantum computer is ***`inherent`***. And it is not outside parallelism which has trust in us by having several processors.

4. `Measurement and Quantum Interference`:
   - In quantum computation, after performing operations on a qubit or a quantum register, when a measurement is made, the superposition **collapses to a definite classical state**.
   - Quantum interference effects, resulting from the **superposition** and **manipulation of qubits**, allow for complex calculations, optimizations, and algorithms that exploit interference patterns among possible states to yield the desired output efficiently.
   - In other words, we compute the value of the function for each one of the inputs at the same time.

In summary, the fundamental difference between classical and quantum computation lies in the ability of qubits to exist in **superposition**, enabling computations on multiple inputs simultaneously through inherent quantum parallelism, *without relying on external parallel processing units.* This unique property forms the basis for the potential computational advantages of quantum computers for certain problems over classical computers.

4.4 Miniaturization Challenges and Landauer's Principle :

Now when miniaturization proceeds like this `(according to Moore's law)` there are **two** problems associated with, when the separation between different components reach atomic dimensions. We know that from the quantum mechanical `uncertainty principle` due to `Heisenberg`, and that has a lot of influence on what happens when things reach atomic dimension.

1. In other words, if the components come so close then the results that you get out of that computation will no longer be reliable.
2. The other thing that will happen is that the **heat** produced by one of the components would naturally `affect the performance` of nearby components and so therefore, this will also make the computation unreliable.

This heat problem has certain other aspects, for instance, the heat produced by a computer depends on the **volume occupied** by the number of bits. But it is required to remove the heat continuously and heat can be removed only from the surface. So, as a result, when the components come too close, the `efficiency` of removing heat will not be quite as good.

Landauer Principle : The Landauer Principle, proposed by ***Rolf Landauer*** in the 1960s, is a fundamental concept in the field of computational physics and information theory. It establishes a

relationship between `information processing`, specifically erasure of information, and `physical thermodynamics`.

The principle can be summarized as follows:

The erasure of information in a computational process is accompanied by an inevitable minimum amount of energy dissipation, which results in an increase in entropy in the environment.

Key points regarding the Landauer Principle:

1. **Information and Entropy**:
    - In information theory, erasing information generates **entropy**. When information is erased or reset in a computational process, the uncertainty or randomness in the system increases, which corresponds to an increase in entropy.
2. **Thermodynamic Connection**:
    - Landauer linked the act of `irreversible information erasure` (like resetting a bit in a computer) to `physical thermodynamics`. He showed that to erase information irreversibly, some minimal amount of energy dissipation is inevitable.
    - The energy associated with the erasure process is dissipated as heat into the environment, leading to an **increase** in the entropy of the environment.
3. **Energy Consumption**:
    - Landauer's insight implies a fundamental connection between information theory and thermodynamics, stating that there is a *theoretical minimum amount of energy consumption required to erase one bit of information.*
    - This theoretical minimum energy is given by the *product of the temperature of the environment and the increase in entropy caused by the erasure of the bit.*
4. **Implications**:
    - The Landauer Principle has implications for the energy efficiency of computing devices. It sets a **theoretical limit** on the amount of energy needed to perform **irreversible operations** like resetting bits in a computer's memory.
    - In the design of ultra-low-power computing devices and in the development of reversible computing (where information is manipulated without loss), the Landauer Principle's constraints and implications are taken into consideration.

In essence, the Landauer Principle provides a crucial link between information theory and thermodynamics, highlighting the physical costs associated with information processing, particularly in terms of energy dissipation and entropy increase when information is erased irreversibly.

4.5 Reversibility :

AND gate is an irreversible process, & most of the processes in classical computing are done irreversibly.

The Landauer principles state that every ***n bit of information increases the thermodynamic entropy by nk log2**, which would mean that there is a certain amount of **loss of energy** and the

process becomes **gradually inefficient** as the number of components increases. The present-day computers dissipate much more energy than this limit.

So, the quantum processes, have to be carried out `reversibly`, in fact, the operators which will be performing are unitary operators.

`Reversibility` means every logical step should be capable of getting reversed, which results in negligible energy loss.

Now it is also possible to do classical computing using reversible gates, but then there is always a problem of what we call as **garbage**.

- The `garbage arises` because, for example, using AND gate and want it to be reversible, which means we have to store the inputs continuously, in fact, the only classical gate which is a reversible gate is a **NOT** gate.
- But for all others, if we want the process should be done reversibly, we need to **collect** the inputs which are not later on require. So, that becomes a very big `disposal problem` and it also requires unnecessary storage.

So these are the two primary issues connected with the advent of quantum computers.

5 Postulates of Quantum Mechanics :

There are `6` Postulates of Quantum Mechanics.

5.1 Postulate - 1 :

The state of the Quantum-mechanical system is completely specified by the function $\Psi(r,t)$ that depends on the coordinates of the particle, `r` and the time, `t`.

This function is called the **Wave-Function** or **State-Function** and has the property that $\Psi^*(r,t)\Psi(r,t)\,d\tau$ is the probability that the particle lies in the volumetric element $d\tau$ located at `r` and `t`.

This is the `Probabilistic` interpretation of wavefunction. As a result, the wave function must satisfy the condition that finding the particle somewhere in space is `1` and this gives us the normalisation condition

$$\boxed{\int_{-\infty}^{\infty} \Psi^*(r,t)\Psi(r,t)\,d\tau \;= 1}$$

The other conditions on the wavefunction that arise from the probabilistic interpretation are that it must be **single-valued**, **continuous** and **finite**. Generally, wavefunctions are written with the inclusion of `normalisation constant`.

5.1.1 Explanation :

Let's break down the first postulate of quantum mechanics using a simple example involving a `particle in a one-dimensional box`.

Imagine a particle confined within a box, free to move along a line. The wave function, denoted by $\Psi(x,t)$, describes the state of this particle. It represents the probability amplitude of finding the particle at a particular position `x` at time `t`.

- Think of the wave function $\Psi(x,t)$ as a special way to describe where a particle might be in a box at a given time.
- It's like a mathematical map telling us the likelihood of finding the particle at different spots along this line inside the box.
- The wave function gives us a way to calculate the **chance of finding the particle in a small space**.

Imagine splitting the box into tiny pieces. The probability of finding the particle in one of these tiny pieces is given by $\Psi(x,t)$ times its complex conjugate $\Psi^*(x,t)$. We then add up these probabilities for all these tiny pieces along the entire length of the box.

Now, according to the probabilistic interpretation of the wave function, the probability of finding the particle in a small interval **dx** at a specific time is given by $\Psi(x,t)$ multiplied by its complex conjugate $\Psi^*(x,t)$, and then integrated over that interval:

$$\boxed{\int_x^{x+dx} \Psi^*(x,t)\Psi(x,t)\,dx}$$

- For simplicity, let's say our box spans from `x = 0` to `x = L`, so the particle can only exist within this region.
- For the particle to exist somewhere within the entire box at time `t`, the integral of the probability density over the entire space must equal 1:

$$\implies \int_0^L \Psi^*(x,t)\Psi(x,t)\,dx = 1$$

To know where the particle is in the whole box, we add up these probabilities for every possible position within the box, from x = 0 to x = L. `The total of all these probabilities should always add up to 1`.

This means the particle is definitely somewhere in the box—it's just a matter of figuring out where! This condition ensures that there is a 100% chance of finding the particle somewhere within the box.

As an example, consider a simple wave function for a particle in this box:

$$\Psi(x,t) = A\sin\left(\frac{n\pi x}{L}\right)$$

Here, `A` is a normalization constant and `n` is an integer (1, 2, 3, ...), representing different possible states of the particle.

To satisfy the normalization condition, the constant `A` is chosen such that when we square this wave function ($\Psi^*(x,t)$ times $\Psi(x,t)$) and integrate it over the entire length of the box from 0 to `L`, we get 1.

When we use a specific wave function like $\Psi(x,t) = A\sin\left(\frac{n\pi x}{L}\right)$, the constant A is what we adjust.

We want the total probability to be `1`, so A is chosen in such a way that when we square this wave function and add it up from `x = 0` to `x = L`, the result equals 1.

This normalization constant `A` ensures that the total probability of finding the particle within the box is indeed 1, meeting the requirement of the first postulate of quantum mechanics.

5.1.2 Example of finding treasure on the planet:

Let's imagine applying the concept of the wave function's probabilistic interpretation to locating a particular point on the Earth's surface using latitude and longitude coordinates.

Scenario:

- Consider a treasure hunt where a hidden treasure represents a particle in the quantum world.
- The probability of finding the treasure at any given spot on Earth is akin to the probability of locating a particle in a specific volume element in quantum mechanics.

Equating to Wave Function Probability:

- The Earth's surface is represented by latitude and longitude coordinates.
- Each point on the Earth corresponds to a specific coordinate (`r`).

Quantum Mechanics Interpretation:

- In quantum mechanics, the square of the wave function at a particular point represents the probability of finding the particle at that position.
- Similarly, the probability of finding the treasure at a particular geographic location corresponds to the probability density at that specific lat-long coordinate.

Normalization Condition:

- Ensuring that the treasure exists somewhere on Earth is equivalent to the quantum mechanics normalization condition.
- Just as the total probability of finding the particle across all space is `1`, in our treasure hunt scenario, the treasure must be located somewhere on Earth.

Application:

- Using the analogy of the wave function, the treasure's probable locations are spread across the globe, each with its probability density.
- The quest for the treasure is similar to exploring the quantum state space, searching for the highest probability location.

This analogy demonstrates how the wave function's probabilistic nature in quantum mechanics aligns with the probabilistic interpretation of finding an object (like the treasure) at different locations on Earth.

In quantum mechanics, the wave function's properties dictate the probability distribution of a particle's location, much like how the treasure's location is probabilistically spread across the Earth's surface.

5.2 Postulate - 2 :

To every Observable in the classical mechanics there corresponds a linear, Hermitian operator in Quantum mechanics.

This Postulate comes from the observation that the expectation value of an operator that corresponds to an observable must be real and therefore the operator must be Hermitian. Some example of Hermitian Operators are :

Observable	Classical Symbol	Quantum Operator	Operation
Position	$\mathbf{r}$	$\hat{r}$	Multiply by $\mathtt{r}$
Momentum	$\mathbf{p}$	$\hat{p}$	$-i\hbar(\hat{i}\frac{\partial}{\partial x} + \hat{j}\frac{\partial}{\partial y} + \hat{k}\frac{\partial}{\partial z})$
Kinetic Energy	$\mathbf{T}$	$\hat{T}$	$\frac{-\hbar^2}{2m}(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2})$
Potential Energy	$\mathbf{V(r)}$	$\hat{V}(r)$	Multiply by V(r)
Total Energy	$\mathbf{E}$	$\hat{H}$	$\frac{-\hbar^2}{2m}(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}) + V(r)$
Angular Momentum	$\mathbf{l_x}$	$\hat{l_x}$	$-i\hbar(y\frac{\partial}{\partial z} - z\frac{\partial}{\partial y})$
	$\mathbf{l_y}$	$\hat{l_y}$	$-i\hbar(z\frac{\partial}{\partial x} - x\frac{\partial}{\partial z})$
	$\mathbf{l_z}$	$\hat{l_z}$	$-i\hbar(x\frac{\partial}{\partial y} - y\frac{\partial}{\partial x})$

5.2.1 Explanation :

The second postulate of quantum mechanics connects classical observables (like position, momentum, and energy) to their corresponding quantum operators. Here's a simplified explanation:

Observable Correspondence:

- This postulate establishes a connection between classical observables (like position, momentum, energy) and their counterparts in quantum mechanics.
- For every measurable property in classical physics, there exists a corresponding quantum operator in quantum mechanics.

Hermitian Operators:

- These quantum operators must be Hermitian, meaning they satisfy a specific mathematical property.
- Hermitian operators in quantum mechanics correspond to observable quantities whose measured values are always real.

Role of Operators:

- In quantum computing, these Hermitian operators are crucial elements. They represent the fundamental properties of quantum systems that we want to measure or manipulate during computations.

Example with Quantum Gates:

- Quantum gates in quantum computing, like the Pauli-X gate or the Hadamard gate, correspond to specific operators.
- These gates (operators) perform operations on quantum bits (qubits) to execute quantum algorithms.

Observables in Quantum Algorithms:

- The classical observables' counterparts in quantum mechanics are employed as operators in quantum algorithms.
- For instance, the quantum phase estimation algorithm uses the phase as an observable, represented by a specific quantum operator.

Real-Life Analogy:

- Consider classical tools used to measure properties of objects. In quantum computing, the Hermitian operators are like the mathematical tools that allow us to make computations and measurements within the quantum realm.

Practical Impact:

- Hermitian operators' used in quantum computing forms the backbone of quantum algorithms and computations.
- They enable us to simulate and analyze quantum systems and perform computations beyond the capabilities of classical computers.

In essence, the second postulate underlines the connection between classical observables and their quantum mechanical counterparts, emphasizing the importance of Hermitian operators in quantum computing. These operators are pivotal in designing algorithms and performing computations in the quantum realm.

5.3 Postulate - 3 :

In any measurement of the observable associated with operator $\hat{\mathbf{A}}$, the only values that will ever be observed are the `eigenvalue`, `a`, that satisfy the eigenvalue equation,

$$\boxed{\hat{A}\Psi = a\Psi}$$

This is the postulate that the values of dynamical variables are quantized in Quantum mechanics (although it is possible to have a continuum of eigenvalues in the case of unbound states). If the system is in an eigenstate of $\hat{A}$ with eigenvalue $\boldsymbol{a}$ then any measurement of the quantity $\boldsymbol{A}$ will always yield the value $\boldsymbol{a}$.

Although measurement will always yield a value, the initial state does not have to be an eigenstate of $\hat{A}$. An arbitrary state can be expanded in the complete set of Eigenvectors of $\hat{A}$, $\hat{A}\Psi_i = a_i\Psi_i$,

as

$$\boxed{\Psi = \sum_{i}^{n} c_i \Psi_i}$$

- where n may go to infinity(∞).

In this case measurement of $\boldsymbol{A}$ will yeild one of the eigenvalues, a_i but we don't know which one. The *probability* of observing the eigenvalue a_i is given by absolute value of the square of the coefficient, $|c_i|^2$.

The third Postulate also implies that, after the measurement of Ψ yields some value, a_i, the wavefunction ***collapses*** into eigenstate Ψ_i that corresponds to a_i. If a_i is degenerate Ψ collapses onto the degenerate subspace. Thus the act of measurement affects the state of the system and this has been used in many elegant experimental explorations of Quantum mechanics (ex: Bell's Theorem).

5.3.1 Explanation :

The third postulate of quantum mechanics, often referred to as the **postulate of measurement** or **the collapse of the wave function**, has significant implications for quantum computing.

Quantization of Observable Values:

- In quantum mechanics, when measuring an observable represented by an operator $\hat{A}$, the only values observed are the eigenvalues a that satisfy the equation $\hat{A}\Psi = a\Psi$.
- This implies that the values of physical properties are quantized in quantum systems. For instance, if a system is in an eigenstate of $\hat{A}$ with eigenvalue a, any measurement of that property will yield the value a.

Eigenstates and Probabilities:

- Although the initial state may not be an eigenstate of $\hat{A}$, it can be expanded into a set of eigenvectors of $\hat{A}$ $\hat{A}\Psi_i = a_i\Psi_i$ using coefficients (c_i).
- The probability of observing the eigenvalue (a_i) upon measurement is given by the absolute value squared of the coefficient $(|c_i|^2)$.
- This means that the measurement outcome is probabilistic, with each eigenvalue having a probability associated with it.

Wave Function Collapse:

- After the measurement, the state of the system changes. The wave function `collapses` into the eigenstate Ψ_i corresponding to the observed eigenvalue (a_i).
- If (a_i) is degenerate (multiple states with the same eigenvalue), the wave function collapses into the degenerate subspace, affecting the state of the system.

Implications in Quantum Computing:

- In quantum computing, this postulate implies that measurements collapse the quantum state to a specific outcome.
- It underscores the probabilistic nature of measurement outcomes in quantum systems.

- Quantum algorithms leverage this probabilistic nature and exploit superposition and entanglement to perform computations more efficiently for certain problems.

In essence, the act of measurement in quantum computing causes the system to collapse into a specific state, impacting subsequent computations or measurements. This phenomenon is fundamental and central to quantum algorithms and their unique computational power.

5.3.2 Quantum Coin Example:

Let's consider a theoretical scenario with a quantum coin that can be in a superposition of both heads (`H`) and tails (`T`). When measured, the quantum coin collapses to either `H` or `T` with a certain probability associated with each outcome.

Imagine a quantum coin that, when flipped, is in a superposition of `H` (heads) and `T` (tails) states simultaneously. This coin represents a qubit in a quantum state.

1. **Superposition:** The quantum coin is in a superposition state $|\Psi\rangle = \frac{1}{\sqrt{2}}|H\rangle + \frac{1}{\sqrt{2}}|T\rangle$, meaning it's equally likely to be in both states `H` and `T` simultaneously.
2. **Measurement:** When observed or measured, the quantum coin collapses into either `H` or `T` based on probabilities.
   - The probability of observing `H` is $\left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$.
   - Similarly, the probability of observing `T` is $\left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$.
3. **Outcome:** Upon measurement, the quantum coin will yield either `H` or `T`. However, before measurement, it exists in a superposition of both.

Real-Time Analogous Scenario:

Imagine you have a hypothetical quantum coin that behaves as described above. You flip this quantum coin, but instead of observing the result immediately, you're blindfolded and don't see the outcome until later.

While blindfolded, the coin exists in a superposition of `H` and `T`, just like a qubit in a quantum state. Only when you remove the blindfold and observe the result does the quantum coin "***collapse***" into either `H` or `T`.

The probabilistic nature of quantum mechanics means that, until you observe the coin, it effectively remains in a simultaneous `H` and `T` state. The act of observation (removing the blindfold) causes the collapse, revealing the actual outcome.

This analogy helps visualize the probabilistic nature of quantum measurements. Until observed, the quantum system exists in multiple states simultaneously, demonstrating the essence of superposition and the impact of observation on the final outcome.

5.4 Postulate - 4 :

If a system is in a state described by the normalised wavefunction, Ψ, then the average value of the observable corresponding to $\hat{A}$ is given by

$$\boxed{\langle \hat{A} \rangle = \int_{-\infty}^{+\infty} \Psi^* \hat{A} \Psi \, d\tau}$$

5.4.1 Explanation :

The fourth postulate of quantum mechanics addresses how to calculate the average or expected value of an observable associated with a quantum system, represented by the wave function Ψ.

Calculating the Average Value of an Observable:

- For a system described by the normalized wave function Ψ, the average value of an observable represented by the operator $\hat{A}$ is calculated as:

$$\boxed{\langle \hat{A} \rangle = \int_{-\infty}^{+\infty} \Psi^* \hat{A} \Psi \, d\tau}$$

where Ψ^* denotes the complex conjugate of the wave function.

In Simple Terms:

- Consider a quantum system described by the wave function Ψ. This wave function contains information about the system's state, such as its **position**, **momentum**, or **other observables**.
- The average or expected value of an observable (like position, energy, etc.) represented by the operator $\hat{A}$ in this quantum system is calculated using an **integral**.
- The integral involves the **complex conjugate of the wave function** Ψ^*, the **operator** $\hat{A}$, and the **original wave function** Ψ.
- This calculation gives us the average value that we would expect to measure if we were to perform measurements on the quantum system multiple times.

Practical Implications:

- This postulate helps in predicting the expected outcomes of measurements on a quantum system.
- By calculating the average value of observables, it provides insight into the behavior of the system and what outcomes are likely to be observed in experiments.

In essence, the fourth postulate provides a mathematical framework to determine the expected or average outcome of measuring an observable in a quantum system described by its wave function. It's a fundamental aspect of quantum mechanics used to predict and understand measurement outcomes in the quantum world.

5.4.2 Particle in a box Example :

Let's consider a simple real-life example involving a quantum particle's position to understand the fourth postulate.

Finding Average Position of a Particle in a Box:

Imagine a particle confined within a **one-dimensional box**, represented by a wave function $\Psi(x)$. We want to find the average position of this particle within the box using the fourth postulate.

1. **System Description:**

- The wave function $\Psi(x)$ describes the **probability amplitude** of finding the particle at a specific position x inside the box.

2. **Calculating Average Position:**
   - Using the fourth postulate, the **average position** $\langle x \rangle$ of the particle is determined by the integral:

$$\boxed{\langle x \rangle = \int_{-\infty}^{+\infty} \Psi^*(x) \hat{x} \Psi(x)\, dx}$$

   where $\hat{x}$ is the **position operator**.

3. **Interpretation:**
   - This calculation yields the average or expected value of the particle's position if we were to perform measurements on its location multiple times.
   - The result provides insight into **where the particle is more likely to be found within the box** based on the given wave function.

Real-Life Analogy:

Think of this scenario as trying to predict where a tiny particle might be inside a confined box without directly observing it. The wave function $\Psi(x)$ acts like a **probability map**, indicating where the particle is more likely to be found within the box.

- Using the fourth postulate is akin to calculating the average position of the particle inside the box based on this probability map.
- While we can't directly see the particle's position, this calculation helps us estimate where it's most likely to be found on average, providing a prediction based on the quantum properties described by the wave function.

In summary, the fourth postulate allows us to estimate or predict the expected outcome (such as the average position of a quantum particle) based on the mathematical description of the system provided by the wave function, even without directly observing the particle.

5.5 Postulate - 5 :

The wavefunction or state function of a system evolves in time according to the time-dependent Schrodinger equation

$$\boxed{\hat{H}\Psi(r,t) = i\hbar \frac{\partial \Psi}{\partial t}}$$

5.5.1 Explanation :

The fifth postulate of quantum mechanics deals with how the wave function or state function of a quantum system changes over time. It's described by the time-dependent Schrödinger equation.

Evolution of the Wave Function:

- The **time-dependent Schrödinger equation** represents how the wave function $\Psi(r,t)$ of a quantum system changes over time.

$$\hat{H}\Psi(r,t) = i\hbar \frac{\partial \Psi}{\partial t}$$

where $\hat{H}$ is the **Hamiltonian operator**, representing the *total energy of the system.*

Explanation in Simple Terms:

- Consider a quantum system described by its wave function $\Psi(r,t)$. This wave function contains information about the system's state at different positions r and at time t.
- The **Schrödinger equation** describes how this wave function evolves over time. It states that the total energy operator $\hat{H}$ acting on the wave function produces the *rate of change of the wave function with respect to time.*

Interpretation:

- The equation essentially governs how the state of a quantum system changes over time. It links the energy of the system (through the Hamiltonian operator) to the time evolution of its wave function.

Practical Implications:

- Solving the time-dependent Schrödinger equation helps predict the behavior of quantum systems as they evolve over time.
- It's fundamental for understanding how quantum states change and develop, aiding in the prediction of outcomes in experiments and real-world applications.

In essence, the fifth postulate describes the evolution of a quantum system's state function over time. It's a fundamental equation that underpins the dynamics of quantum systems and allows us to predict how their states change with time, providing crucial insights into the behavior of quantum systems.

5.5.2 Example of electron in an Atom:

Let's consider a real-life example involving the time evolution of an electron in an atom.

Scenario:

- An electron in an atom is described by a wave function $\Psi(r,t)$, where (r) represents its position and (t) represents time.
- The time-dependent Schrödinger equation governs how this wave function changes over time.

Explanation:

- The Schrödinger equation for the electron in the atom $\hat{H}\Psi(r,t) = i\hbar\frac{\partial\Psi}{\partial t}$ describes the evolution of the electron's wave function.
- The Hamiltonian operator $\hat{H}$ encapsulates the total energy of the electron within the atom.
- Solving this equation helps predict how the electron's probability density distribution, described by the wave function $\Psi(r,t)$, changes as time progresses.

Real-Life Analogy:

- Think of the **electron's behavior as a dance within the atom**, where its movement and position are represented by the changing wave function.

- The Schrödinger equation acts like the choreography, dictating how the electron's "dance" evolves over time. It guides the electron's movements within the atom, determining its probabilities of being in different positions at different times.

Practical Implications:

- Understanding the time evolution of quantum systems, like the electron in an atom, helps predict their behavior and properties at various points in time.
- This knowledge is crucial in fields such as quantum chemistry, materials science, and quantum computing, where predicting the behavior of quantum particles over time is essential for designing new materials or understanding chemical reactions.

In summary, the time-dependent Schrödinger equation allows us to model and predict the evolution of quantum systems over time, providing valuable insights into the behavior of particles such as electrons in atoms in various real-world applications.

5.6 Postulate - 6 :

The total wavefunction must be **antisymmetric** with respect to the interchange of all coordinates of one *fermion* with those of another. Electron spin must be included in this set of coordinates.

- The Pauli execlusion principle is a direct result of this *antisymmetry* postulate

5.6.1 Explanation :

The sixth and final postulate of quantum mechanics pertains to the **antisymmetric nature of the total wave function** for systems of identical particles, specifically fermions like electrons. This leads directly to the Pauli Exclusion Principle.

Antisymmetric Wave Function:

- For systems of identical particles (fermions), the total wave function describing their states must be ***antisymmetric*** under the interchange of any two particles' coordinates (including electron spin).

Pauli Exclusion Principle:

- The *Pauli Exclusion Principle* is a consequence of this antisymmetric nature.
- It states that no two identical fermions (such as electrons) within a system **can occupy** the same quantum state simultaneously.
- Specifically, it prohibits two identical fermions from having the same set of quantum numbers (spin, orbital, etc.) in a given system.

Explanation in Simple Terms:

- Imagine a system of identical particles (like electrons) described by a total wave function.
- The wave function must change sign (become negative) when the coordinates of any two particles are interchanged. This antisymmetry ensures that the wave function remains unchanged under permutations, ensuring a unique behavior for identical particles.

Pauli Exclusion Principle in Practice:

- The Pauli Exclusion Principle has profound implications, especially in atomic and molecular systems.
- It limits the number of electrons that can occupy a given energy level or orbital in an atom, leading to the stability and structure of atoms and the periodic table.
- It prevents the collapse of matter into infinitesimally small volumes by prohibiting too many particles from occupying the same state.

5.6.2 Example of rule at a party :

Think of the Pauli Exclusion Principle as a rule at a party:

- Imagine a dance floor representing quantum states, and individuals (electrons) with their unique dance moves (quantum properties).
- The rule states that no two individuals can perform the exact same dance moves simultaneously on the same spot on the dance floor. This ensures diversity and avoids overcrowding in certain spots.

Practical Importance:

- The Pauli Exclusion Principle is crucial in understanding the stability of matter, the behavior of electrons in atoms, and the formation of chemical bonds.
- It underpins the electronic structure of atoms and molecules, influencing material properties and chemical reactions.

In summary, the sixth postulate's antisymmetric requirement of the wave function for identical particles leads to the Pauli Exclusion Principle, governing the behavior and arrangement of fermions like electrons in quantum systems.

6 Bits V/s Qubits :

6.1 Dirac Notations :

Any classical computer basically functions on two basic units that are 0s and 1s. These represents states that any memory unit the computer can be in. In Quantum Computer the comparision is with Qubits or Quantum bits. Quantum bits can be in states that are loosely equivalent to 0 and 1. These are better considered as unit vectors rather than states.

A Qubit can be more than a 0 or 1, it can be the linear combination of Zero State and one state. This is represented in the below figure where alpha and beta complex numbers, and it is called as **Super-Position** state.

Comparision of Classical Bits Vs Qubits :

Category	Classical Bits	Qubits
States representation	**0** & **1**	$\|0\rangle$ & $\|1\rangle$
States type	Binary Values	Unit Vectors

Category	Classical Bits	Qubits
Usual Representation	Binary strings	Linear combination of unit vectors
	$b = 0, 1$	$\boxed{\|\Psi\rangle \ = \alpha\|0\rangle \ + \beta\|1\rangle}$
Nature of States	Deterministic	Probabilistic
		$\boxed{\|0\rangle = \|\alpha\|^2, \quad \|1\rangle = \|\beta\|^2}$

Ket : The most convienient way of representing the Quantum states is with **Dirac-Notation** or also called as **Bra-ket Notation**.

- **Ket** is basic object in quantum computing and it basically used to represent the Quantum State.
- Roughly, we can consider **ket** as equivalent of orthogonal unit vectors in 2D-Vector Space.

Ket-0 representation: $\boxed{|0\rangle \ = \begin{bmatrix}1\\0\end{bmatrix}}$

Ket-1 representation: $\boxed{|1\rangle \ = \begin{bmatrix}0\\1\end{bmatrix}}$

Generally any state of a particle $|\Psi\rangle$ is represented as linear combination of $|0\rangle$ and $|1\rangle$. It is given by

$$\boxed{|\Psi\rangle \ = \alpha|0\rangle \ + \ |1\rangle}$$

The same can be imagined as the complex number combinations for vector space and for Linear algebra format the α & β constants are actually an elements in **Complex Hilbert Space** or **Vector Space**, and this Ket is a **Linear Combination of two OrthoNormal basis States**.

Dirac Notation	vector Notation	Matrix Notation
$\|0\rangle \ \|1\rangle$	$\hat{i}\&\hat{j}$ are **orthogonal vectors**	$\boxed{\|0\rangle \ = \begin{bmatrix}1\\0\end{bmatrix}}$ and $\boxed{\|1\rangle \ = \begin{bmatrix}0\\1\end{bmatrix}}$
$\boxed{\|\Psi\rangle \ = \alpha\|0\rangle \ + \ \|1\rangle}$	$\alpha\hat{i} \ + \beta\hat{j}$ where α & β are complex numbers.	$\begin{bmatrix}\alpha\\\beta\end{bmatrix}$ elements of Complex Hilbert-Space this matrix forms linear combination of orthonormal basis states

Hilbert Space : It is a complex multidimenstional vector space where inner product of any pair of elements is defined as extrapolation of a 2D Space to n-Dimensions and Complex coefficients.

- $|0\rangle$, $|1\rangle$ are considered as the standard basis states also as orthonormal basis states.

- Consider a 2D plane with an x-axis and y-axis and we have a unit vector that aligns with x-axis and a unit vector that aligns with y-axis.
- Now we can represent any point on 2D plane as the linear combination of unit vectors of x and y.
- Hilbert space is similar to to it. Here the coefficients are complex numbers so the mathematics remains very similar.
- It is just like we are having different kind of space and, standard basis roughly equivalent to our **i & j** vectors on a 2D plane which they represent **orthonormal basis states**.(They are perpendicular to each other).

Bra : It is the opposite operation of **Ket**($|\rangle$;).

- It is actually an operation or a function that can be applied on Ket.
- **For a given Ket we can compute Bra as the conjugate Transpose of Ket**

Bra-0 representation : $\boxed{\langle 0| = [conjugate\ of \begin{bmatrix}1\\0\end{bmatrix}]^T \implies \langle 0| = \begin{bmatrix}1\\0\end{bmatrix}^\dagger}$

Bra-1 representation : $\boxed{\langle 1| = [conjugate\ of \begin{bmatrix}0\\1\end{bmatrix}]^T \implies \langle 1| = \begin{bmatrix}0\\1\end{bmatrix}^\dagger}$

- If **A** is any **ket** $|A\rangle$; then its bra is given by $\langle A|$ which is

$$\boxed{\langle A| = A^\dagger = [\overline{A}]^T}$$

and complex conjugate means if $\alpha = a + ib$ then, α^* *or* $\alpha^\dagger = a - ib$.

Bra is represented as **row vector** and Ket is represented as **column vector**.

$$\boxed{\begin{bmatrix}\alpha\\\beta\end{bmatrix} \implies \begin{bmatrix}\alpha^* & \beta^*\end{bmatrix}}$$

At N dimensional scale : Now let us consider vectors with n dimensions.

- Consider a general **ket-Si** which is actually a **nx1** matrix.

$$\boxed{|\Psi\rangle = \begin{bmatrix}a_1\\a_2\\.\\.\\.\\a_n\end{bmatrix}}$$

- Similarly a general **bra-Phi** which is actually a **1xn** matrix.

$$\boxed{\langle\Phi| = \begin{bmatrix}b_1^* & b_2* & . & . & . & b_n*\end{bmatrix}}$$

Bra-Ket Operation : This Operation applies **bra-phi** to **Ket-si** and this is an inner product of both the vectors or better can be interpreted as the Dot product of both the vectors.

$$bra-ket\langle\Phi|\Psi\rangle = \begin{bmatrix} b_1^* & b_2^* & . & . & . & b_n* \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_n \end{bmatrix} = b_1^* a_1 + b_2^* a_2 + ... + b_N^* a_N$$

Here the result is the scalar product.

If we consider $|0\rangle$ & $\langle 1|$ and performed $\langle \mathbf{1}|\mathbf{0}\rangle$ Bra-Ket operation then the result is as follows :

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

- If the **inner product** of any two **unequal** vectors (i.e bra-ket operation of two vectors) is **0** then the vectors are **Orthogonal** to each other.
- The inner product of any two **same** vectors is always **1** which is the indication for the **normalization**. All quantum states are always normalized.

Ket-Bra Operation : (Kronecker Product) This operation is similar to Bra-Ket but here it performs the outer product instead of inner product. It is done as follows:

$$Ket-Bra\ |\Psi\rangle\langle\Phi| = \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_n \end{bmatrix} \begin{bmatrix} b_1^* & b_2* & . & . & . & b_n* \end{bmatrix} = \begin{bmatrix} a_1 b_1^* & a_1 b_2^* & . & . & . & a_1 b_N* \\ a_2 b_1^* & a_2 b_2^* & . & . & . & a_2 b_N* \\ . & & & & & \\ . & & & & & \\ . & & & & & \\ a_N b_1^* & a_N b_2^* & . & . & . & a_N b_N* \end{bmatrix}$$

If we consider $|0\rangle$ & $\langle 1|$ and performed $\mathbf{|0\rangle\langle 1|}$ Bra-Ket operation then the result is as follows :

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

Kronecker Product of two Matrices: Consider two matrices **A** and **B** as shown below:

$$\begin{bmatrix} a11 & a12 \\ a21 & a22 \end{bmatrix} \begin{bmatrix} b11 & b12 \\ b21 & b22 \end{bmatrix}$$

- The product of **A & B** is:

$$\begin{bmatrix} a11 * B & a12 * B \\ a21 * B & a22 * B \end{bmatrix} = \begin{bmatrix} a11 * b11 & a11 * b12 & a12 * b11 & a12 * b12 \\ a11 * b21 & a11 * b22 & a12 * b21 & a12 * b22 \\ a21 * b11 & a21 * b12 & a22 * b11 & a22 * b12 \\ a21 * b21 & a21 * b22 & a22 * b21 & a22 * b22 \end{bmatrix}$$

6.2 For Single Qubit :

A Single qubit is nothing but a vector in a complex 2D-Hilbert Space.

A general Qubit state is represented as Arbitary Ket which is linear combination of orthonormal basis state 0 and with complex Coefficients. It is given as follows :

$$\boxed{|q\rangle \ = \alpha|0\rangle \ + \beta|1\rangle}$$

$$where|q\rangle \implies \begin{bmatrix}\alpha\\ \beta\end{bmatrix} \ and \ \alpha, \beta \ \epsilon \ C(complexnumbers)$$

$$|0\rangle \ and \ |1\rangle \ are \ Ortho-Normal \ Basis \ States(\ inner \ Product \ = 0)$$

The matrix representation of this is a **2x1** matrix Which represent the Quantum States. Generally they can be described as a linear combination of basis states, as well as a superposition of **0** and **1**.

When we actually look at those qubits, they **probabilistically collapse to either of those basis states** so they do not remain in same form when we look at them. So they can be assumed as either 0 or 1 which means that being in state 0 or state 1 happens to be definite.

Consider the following example where we need to find the probability of qubit q in state 0 and state 1.

Probability of qubit in the standard basis $\{|0\rangle, |1\rangle\}$

$$\boxed{\begin{aligned} P(|q\rangle = |0\rangle) &= |\langle 0|q\rangle|^2 = |\langle 0|(\alpha|0\rangle + \beta|1\rangle)|^2 = |\alpha\langle 0|0\rangle + \beta\langle 0|1\rangle|^2 = |\alpha|^2 \\ P(|q\rangle = |1\rangle) &= |\langle 1|q\rangle|^2 = |\langle 1|(\alpha|0\rangle + \beta|1\rangle)|^2 = |\alpha\langle 1|0\rangle + \beta\langle 1|1\rangle|^2 = |\beta|^2 \end{aligned}}$$

$$\boxed{|\alpha|^2 + |\beta|^2 = 1}$$

$$\boxed{\begin{aligned} &\text{Since Complex number} \implies \alpha = a + ib \text{ or } re^{i\theta} \\ &\text{Modulus} \implies |\alpha| = \sqrt{a^2 + b^2} \end{aligned}}$$

First conider the probability of qubit q in state 0. So initally we need to find product of bra-0 and Ket-q, we will get any complex scalar as the result.

- So we need to apply modulus to find the magnitude of the scalar and apply square on it. But we have considered the ket-q as the linear combination of ket-0 and Ket-1 with coefficients alpha and Beta.
- So we need to expand the expression of bra-ket of 0 and q.

As all the operations we are performing are linear transformations only, we can take all the distribute all the scalar parts in the expression. Here we make product of bra-0 with Ket-0 and ket-1 by taking coefficients common.

We know that Bra-ket 0-0 is 1 and Bra-Ket 0-1 is 0. So we will remain with $|\alpha|^2$. Similarly we can find the probability of qubit q being in state 1.

We know that sum of all probabilities is equals to 1. So the sum of the results of probabilities of q being in state-0 and state-1 is equals to **1**.

This state of qubits is called **normalized state of qubits** which are represented as **bra-ket of q-q**.The coefficients Alpha & Beta are also called as **Probability Amplitudes** for a general qubit.

$$\boxed{\langle \mathbf{q}|\mathbf{q}\rangle = \mathbf{1} \implies \text{Normalized State}}$$

6.3 Ray representation of Physical States :

Physical states are represented by **Rays** in **Hilbert Space (H)**. These vectors in Hilbert space are the represented as **Kets**.

- A Hilbert space is a **normed vector space** in which the concept of an inner product is defined.
- Usually this space is **infinite dimensional** and it has another property that the space is **complete** in the sense it is closed.

However in quantum computing we normally deal with finite vector spaces as a result of which the closedness or the completeness of the vector space is automatically satisfied.

The inner product as we will see is very much like a **scalar product of two vectors with some minor differences**.

And the concept of a norm, that is a length essentially is defined in this space. So this is an abstract vector space in which we define vectors or the vectors represent the physical states of the system.

- $|\Psi\rangle$ is the vector standing for a physical system in this space.

Now the corresponding to this space there is a **dual space** which is represented by $\langle|$. As we know that $\langle|\rangle$ operation is the inner product of two vectors they are not same as Dot Product. - This inner Product is defined in this space which is linear. If we look at the inner product of the bra corresponding to Ψ with the vectors Ψ itself now this is **greater than or equal to 0** and it becomes zero only when $|\Psi\rangle$ is a null vector.

$$\boxed{\begin{aligned} \textbf{Dual-Space} &\implies |\Psi\rangle \equiv c|\Psi\rangle \; and \; \langle\Psi| \equiv c^*\langle\Psi| \\ \textbf{Positivity} &\implies \langle\Psi|\Psi\rangle \geq 0 \text{ [equal only if ket is null]} \\ \textbf{Linearity} &\implies \langle\phi_1|a\psi_1 + b\psi_2\rangle = a\langle\phi_1|\psi_1\rangle + b\langle\phi_1|\psi_2\rangle \end{aligned}}$$

In Linear vector space

$$\langle\psi|a\phi_1 + b\phi_2\rangle = a\langle\psi|\phi_1\rangle + b\langle\psi|\phi_2\rangle$$

$$\langle\psi|\phi\rangle^* = \langle\phi|\psi\rangle$$

$$\langle\psi|\psi\rangle = 1 \implies \text{Normalization}$$

Now we know that Physical systems are often referred as Rays in Space. Now this is because in reality there is no difference between a state $\mathbf{C}|\Psi\rangle$ and $|\Psi\rangle$ where C is a general complex number.

Now as a result $\mathbf{C}|\Psi\rangle$ **and** $|\Psi\rangle$ represents the same physical states and so therefore instead of a physical system being represented by a particular vector in the Hilbert space it is represented by a one-dimensional subspace of the Hilbert space for which any multiplication of the vector $|\Psi\rangle$ gives me the same, it represents the same physical state.

6.4 Basis States :

Basis:

If the vector space is spanned by $\{|e_n\rangle\}$ where $n = 1, 2, ..., d$

$$|\psi\rangle = \sum a_n |e_n\rangle$$

The basis may be chosen orthonormal

$$\langle e_m|e_n\rangle = \delta_{m,n}$$

$$a_n = \langle e_n|\psi\rangle$$

- A basis refers to a set of vectors that are linearly independent (meaning none can be expressed as a combination of the others) and span the entire space (meaning any vector in that space can be expressed as a linear combination of those basis vectors).

For instance, in a three-dimensional space, the standard basis might consist of three vectors: **(1, 0, 0), (0, 1, 0), and (0, 0, 1)**, which span the entire three-dimensional space.

Let us suppose I have a basis in this vector space and let me say that this space is spanned by the vectors which are $|\mathbf{e_n}\rangle$.

Suppose it is **d** dimensional vector space I have spanned it, the kets $|\mathbf{e_n}\rangle$ so that any arbitrary vector can be written at the linear combination of this basis with the coefficients being $_{\mathbf{n}}$.

The difference between a normal vector space and this space is, our space is over a field of complex numbers and so therefore all numbers could be complex so this $_{\mathbf{n}}$ are in general complex.

Here we can choose this basis to be **orthonormal**.

An orthonormal basis is a fundamental concept in linear algebra. **It refers to a set of vectors within a vector space that are both orthogonal and normalized**.

Let's break down the two components:

- ***Orthogonal vectors:*** Vectors are orthogonal when they are perpendicular to each other. In an n-dimensional vector space, two vectors are orthogonal if their dot product (also known as the inner product) is zero. Geometrically, this means the vectors are at right angles to each other.

- ***Normalized vectors:*** A vector is normalized when its length (magnitude) is equal to 1. This involves dividing the vector by its own magnitude, effectively scaling it to unit length. The magnitude of a vector is calculated using the square root of the sum of the squares of its components.

An orthonormal basis combines these properties.

In a vector space, an orthonormal basis is a set of vectors that are **mutually orthogonal** (i.e., any pair of vectors from the set is orthogonal) and each vector is **normalized** (i.e., each vector in the set has a magnitude of 1).

So if i choose the basis to be orthonormal then we can take the definition of basis as $\langle \mathbf{e_m} | \mathbf{e_n} \rangle$ whose inner product is made to be equal to $_{\mathbf{m,n}}$.

So, we can conclude that $_{\mathbf{n}}$ s are given by inner product of $\langle \mathbf{e_n}| \, \rangle$.

6.5 Copenhagen Interpretation :

State $|\Psi\rangle$ ***has a probabilistic interpretation with*** $|\alpha_n|^2$ ***as the probability with which the system would be found in the state*** $|e_n\rangle$***.***

The statement is discussing a fundamental concept in quantum mechanics known as the probabilistic interpretation of quantum states.

Breakdown of the Statement:

1. **"State $|\Psi\rangle$ has a probabilistic interpretation":**
   - In quantum mechanics, the state of a system is described by a quantum state vector, often denoted as $|\Psi\rangle$ (Psi), representing the quantum state of the system.
   - This state vector represents all the information that can be known about the system.
2. **"with $|\alpha_n|^2$ as the probability":**
   - In a quantum system, when it's in a state $|\Psi\rangle$, it can be expressed as a linear combination of basis states, like $|e_n\rangle$ (where n represents different states in the system).
   - The coefficients of these basis states, represented by α_n, are complex numbers known as probability amplitudes.
   - The squared modulus of these probability amplitudes, $|\alpha_n|^2$, gives the probability of finding the system in the corresponding basis state $|e_n\rangle$ upon measurement.
3. **"with which the system would be found in the state $|e_n\rangle$":**
   - The squared modulus of the probability amplitudes $|\alpha_n|^2$ represents the probability of observing the system in a particular state $|e_n\rangle$ when a measurement is performed.
   - Upon measurement, the system "collapses" into one of the possible states $|e_n\rangle$, and the probability of collapsing into a specific state is given by $|\alpha_n|^2$.

In essence, the statement highlights that in quantum mechanics, the state of a system is described probabilistically. The probability of finding the system in a particular state upon measurement is encoded in the squared modulus of the complex probability amplitudes associated with that state in the system's quantum state vector.

Now according to ***Copenhagen interpretation***, the state vector, which I have said is a ray $|\Psi\rangle$, it has only a probabilistic interpretation.

For example, Consider a a physical system and suppose we want to make a measurement of that system. For this we go for measuring energy or angular momentum or whatever the physical

property we want to do.

so therefore, if we want to make a measurement of this system represented by a ket $|\Psi\rangle$ the best we can predict is that what is the probability with which we will get a certain result. This caused a philosophical disagreement and the one of the prime opponent of this interpretation is Albert Einstein.

Imagine we are playing with a toy that's covered by a blanket. We can feel the shape and size of the toy by touching the blanket, but until we uncover it, we don't know its exact appearance.

In the quantum world, particles behave similarly.

- They exist in many possible states at once, like the hidden toy under the blanket. When we measure or observe them, they "choose" a specific state, just like when we reveal the toy by lifting the blanket. Before we measure, the particle is in a superposition of states, and the act of measurement forces it into one definite state.
- The Copenhagen interpretation suggests that at the quantum level, particles don't have definite properties until we measure or observe them.

It also proposes that the act of measurement itself affects the outcome, influencing what state the particle will take. This idea challenges our classical intuition, where we expect things to have precise properties regardless of whether we observe them.

1. **Probabilistic Nature:** The Copenhagen interpretation asserts that in the quantum world, the state of a system is described by a wave function (quantum state vector), denoted as $|\Psi\rangle$ in the statement. This wave function represents all possible states the system can be in simultaneously. However, upon measurement, the system "collapses" into one of these states probabilistically.
2. **Collapse of the Wave Function:** According to the Copenhagen interpretation, when a measurement is made on a quantum system described by the wave function $|\Psi\rangle$, the act of measurement causes the wave function to "collapse" to a single state (one of the possible basis states $|e_n\rangle$) with a probability determined by the squared modulus of the probability amplitudes ($|\alpha_n|^2$).
3. **Observer's Role:** The Copenhagen interpretation emphasizes the role of the observer or measurement apparatus in the process. It states that until measurement occurs, the system exists in a superposition of states, and it's only upon measurement that a specific state is observed, with the probabilities given by the squared moduli of the probability amplitudes.

Therefore, the probabilistic interpretation of quantum states, where the squared moduli of probability amplitudes determine the likelihood of finding a system in a particular state upon measurement, aligns with the central tenets of the Copenhagen interpretation. This interpretation forms the foundation for how quantum mechanics is understood and applied in various experimental contexts.

6.6 Observables and Operators :

Observable (position, momentum, energy etc.) are represented by linear, self-adjoint operators in **Hilbert Space**. An **Operator** acts on a Ket an produces another vector in same Hilbert Space.

$$A|\Psi\rangle = |\Phi\rangle \text{ or } \langle\Psi|B = \langle\chi|$$

Now we have already defined the vector space, now we also need to define operators in that vector space because it is the operations that we will do in a vector space which will take us from one vector to another.

What is an observable?

- As we mentioned that an observable could be things like a position, momentum, energy etc. whatever we would like to measure as property of the particle.
- Now these are represented by, linear, self adjoint operators in the same Hilbert space.

So, basically an operator acting on a given vector of the Hilbert space gives us or takes into another vector in the same space.

From above snapshot we can observe that, $A|\Psi\rangle$ the operator **A** acting on $|\Psi\rangle$, gives us a vector $|\Phi\rangle$. - The operators are linear:

$$A.(a|\Psi\rangle + b|\Phi\rangle) = a(A|\Psi\rangle) + b(A|\Phi\rangle)$$

- Identity operator:

$$I.|\Psi\rangle = |\Psi\rangle$$

- Operators are associative but are not, in general, commutative.
- If there exists an operator B such that

  $ BA=AB=I $ then B is called the inverse of A.

All these operators we are using are oftenly referred to as **Linear Operators**.

What is a Linear Operator? A linear operator means that an operator **A** acting on a linear superposition of two vectors let us suppose it acts on $\alpha|\Psi\rangle + \beta|\Phi\rangle$, α and β are complex numbers.

$$\boxed{\alpha|\Psi\rangle + \beta|\Phi\rangle = \alpha A|\Psi\rangle + \beta B|\Phi\rangle}$$

So this is what a linearity of an operator means.

$$\boxed{\begin{aligned} &\hat{H}(\alpha|\Psi\rangle + \beta|\Phi\rangle) = \alpha\hat{H}|\Psi\rangle + \beta\hat{H}|\Phi\rangle \\ &\implies I|\Psi\rangle = |\Psi\rangle \text{ where} I = \text{Identity} \\ &[A, B] = AB - BA \neq 0 \\ &\implies AB = BA = I \implies B = A^{-1} \end{aligned}}$$

An identity operator acting on any state vector $|\Psi\rangle$ gives me the same vector, it does not do anything actually.

- The point that is to be noted here is that these operators, though they are associative, in the sense. If you add A with B+C or add A+B with C their result is the same.

$$\boxed{(A+B)+C = A+(B+C)}$$

- But these operators in general **do not commute**. So these operators A and B, so in general A, B commutator which is as defined as **AB-BA**, in general, is not equal to zero.
- Now suppose there exists, corresponding to an operator **A** supposing there exists another operator **B**, such that **A times B is equal to B times A is equal to I (the identity)**. Then **B** is called **inverse of A** and is represented as $\mathbf{A^{-1}}$.

$$\boxed{B = A^{-1}}$$

This is the Operator Relationship.

Adjoint Operator :

Adjoint of a vector is commonly represented with Adj or † - Dagger.

Adjoint of an operator:

The dual of the ket $A|\Psi\rangle$ is a bra $\langle\Psi|A^\dagger$

$$\langle\Phi|A|\Psi\rangle = \langle A^\dagger\Phi|\Psi\rangle$$
$$\langle\Psi|A^\dagger\Phi\rangle = \langle(A^\dagger)^\dagger\Psi|\Phi\rangle = \langle A\Psi|\Phi\rangle$$
$$(A^\dagger)^\dagger = A$$

Hermitian Operator :

Every physical variable in the Hilbert space is represented by an operator which are known as self adjoint operator or, also called hermitian operator.

- Special property of an hermitian operator is that it satisfies **Adj(A)=A** or $A^\dagger = A$.

$$\langle\Phi|A\Psi\rangle = \langle A^\dagger\Phi|\Psi\rangle$$
$$(A^\dagger)^\dagger = A$$

Hermitian Operators:

$$A^\dagger = A$$

$$\hat{A} = \frac{|\alpha\rangle\langle\beta|}{|\alpha\rangle\langle\beta|\Psi\rangle = c|\alpha\rangle}$$

Now since we said that an operator acting on a vector gives me another vector in the same space,a general representation of an operator could be of the form that it is a ket followed by a bra i.e $|\alpha\rangle\langle\beta|$. This order of Ket-followed by-Bra is important.

- Let us consider Ket -> $|\alpha\rangle$ and bra -> $\langle\beta|$.

- Let us consider that this ket-bra acts on any state like ket $|\Psi\rangle$ it gives the scalar inner product with the bra-ket operation $\langle\beta|\Psi\rangle$.
- Let this scalar be C. We know that **C times a Vector Ket** $|\alpha\rangle$ doesn't have any different physical significance from that of **Ket** $|\alpha\rangle$.

A complex vector place a space in n dimension is usually represented by C^n. Since we are mostly dealing with 2D this can be represented as C^2. So the Basis in C^2 are:

$$\boxed{\begin{pmatrix}0\\1\end{pmatrix} \text{ and } \begin{pmatrix}1\\0\end{pmatrix}}$$

Any vector in this space can be written as a complex number times the first column vector plus another complex number times the second column vector. But, this is not the unique representation. For instance we can also have a basis with column matrices having **1** and **-1** as elements.

for any C^n

C^2 :

$$\begin{pmatrix}0\\1\end{pmatrix} \text{ and } \begin{pmatrix}1\\0\end{pmatrix}$$

$$\begin{pmatrix}1\\1\end{pmatrix} \text{ and } \begin{pmatrix}1\\-1\end{pmatrix}$$

C^3 :

$$\begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\0\end{pmatrix} \text{ and } \begin{pmatrix}0\\0\\1\end{pmatrix}$$

But the difference is that these two vectors are not normalized where are the previous two vectors are normalized. We can normalize them by finding

$$\boxed{\frac{1}{\text{norm(vector)}} = \frac{1}{\sqrt{2}}}$$

So if a **ket is represented by a column vector** and **bra is represented by a row vector**, this form is essentially what is known as a **matrix direct product**.

Let us consider an example of the following matrix which is the resultant(**A**) of product of Ket and Bra operation.

$$\begin{bmatrix}1+2i & -5i\\ 3i & 4\end{bmatrix}$$

- if we want to find the Adjoint of A then we need to solve **conjugate(A-Transpose)**. A^T is given by

$$\begin{bmatrix}1+2i & 3i\\ -5i & 4\end{bmatrix}$$

- **Conjugate of** A^T is given as:

$$\begin{bmatrix}1-2i & -3i\\ 5i & 4\end{bmatrix}$$

By this we can see that **Adj(A)!=A**.

Let us conider another 3x3 Matrix B.

$$\begin{bmatrix} 3 & 2-i & -3i \\ 2+i & 0 & 1-i \\ 3i & 1+i & 0 \end{bmatrix}$$

Here we can observe some symmetry in the matrix. This makes the **Adj(B) = B**. This means B will be an **Hermitian Matrix**.

Hence, if **A** and **B** are two vectors in hilbert space of Dimension **d** then a ket followed by **A** and a bra followed by **B** has same dimension **dxd**. - Then that matrix product is called as **Kronecker Product or Direct Matrix Product**.

Now we already knew that physical observables such as energy, angular-momentum, momentum etc are represented by self adjoint or Hermitian operator. - According to this postulate, any state $|\Psi\rangle$ has a definite value λ for an observable.

Consider an example we need to find energy of any state $|\Psi\rangle$. Here the observable is Energy.

- Here, we can say that energy has a particular value ,only if, the state $|\Psi\rangle$ happens to be an eigen state of the operator corresponding to energy with an **eigen value** λ.

$$\boxed{\hat{A}|\Psi\rangle = \lambda|\Psi\rangle \text{ where } |\Psi\rangle\text{is eigen vector and } \lambda \text{ is eigen value}}$$

Now an arbitrary state is not necessarily an eigen state of an operator, but the if we take the collection of eigen states corresponding to a given operator in the Hilbert space then this set of eigen states form a complete basis.

What is meant by a complete basis? - it simply means that an arbitrary vector in that state can be expanded in terms of the eigen states of any operator corresponding to whatever observable you have. In other words, I can expand an arbitrary state in terms of, a complete set of eigen states of the energy of it, or I can do the same thing in terms of the complete set of eigen states of let us say angular momentum operator.

Let us consider Energy as a particular observable which we need to find out. So we need to find out eigen states of that energy.

Consider the problem of Hydrogen Atom where we find that energy levels given by **Bohr's Energy conditions** as **-13.6 eV** for ground state, next higher state is having **-13.6/4 eV** and next higher state is having **-13.6/9 eV** etc.

- Now the state corresponding to each one of these energy values are called **Eigen Values**

By this we can say that **arbitrary state of the hydrogen atom can be expanded in terms of these eigen states corresponding to definite eigen values**.

So, according to this second postulate, when we are looking at the measurement of energy, the best we can do is to predict various probabilities with which they will occur. - We cannot surely say that after measuring the energy states of hydrogen atom we will get the value of **-13.6eV**.

Consider the basis of 2D space as discussed before. They we considered vectors of basis as:

$$\begin{bmatrix}1\\0\end{bmatrix} \textit{ and } \begin{bmatrix}0\\1\end{bmatrix}$$

But Let us Consider another basis now which are normalized basis.

$$(1/\sqrt{2})\begin{bmatrix}1\\1\end{bmatrix} \textit{ with } \lambda = 1 \textit{ and } (1/\sqrt{2})\begin{bmatrix}1\\-1\end{bmatrix} \textit{ with } \lambda = -1$$

In other words, in terms of these two bases I can expand any vector in that two dimensional space. Now these vectors are eigen states of a matrix which looks like:

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

$Now (1/\sqrt{2})\begin{bmatrix}1\\1\end{bmatrix}$ *is eigen state of the operator* $\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$

Now so this is an eigen state of this operator with an eigen value let us consider as λ=**+1** and the other one corresponds to λ=**-1**.

- Let us consider both basis vectors as e_1 & e_2.
- Then any operator P_1 corresponding to eigen state e_1 is given by $|e_1\rangle\langle e_1|$.
- Similarly for P_2 eigen state operator is given by $|e_2\rangle\langle e_2|$.

The results of ket-bra operations of the operators can be observed here.

$$P_1 \;=\; |e_1\rangle\langle e_1| \;= \frac{1}{2}\begin{bmatrix}1\\1\end{bmatrix}\begin{bmatrix}1 & 1\end{bmatrix} \;=\; \frac{1}{2}\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}$$

$$P_2 \;=\; |e_2\rangle\langle e_2| \;= \frac{1}{2}\begin{bmatrix}1\\-1\end{bmatrix}\begin{bmatrix}1 & -1\end{bmatrix} \;=\; \frac{1}{2}\begin{bmatrix}1 & -1\\-1 & 1\end{bmatrix}$$

- These P_1 & P_2 are can be considered as **Projection Operator**.

So from spectral theorem we can conclude the following:

$$\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} \;=\; 1*(1/2)\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix} \;+\; -1*(1/2)\begin{bmatrix}1 & -1\\-1 & 1\end{bmatrix}$$

$$=> \boxed{\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix} = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}}$$

6.6.1 Spectral Theorem :

What is a Spectral Theorem?

The Spectral Theorem is a fundamental result in linear algebra that provides a deep understanding of the properties of Hermitian (or self-adjoint) matrices. This theorem reveals crucial insights into their diagonalizability, eigenvalues, and eigenvectors.

Statement of the Spectral Theorem:

Let A be a Hermitian matrix (also known as a self-adjoint matrix). The Spectral Theorem states:

1. **Existence of Eigenvalues:** Every Hermitian matrix has a set of real eigenvalues (possibly repeated).
2. **Orthogonal Eigenvectors:** Corresponding to each eigenvalue, there exists a set of linearly independent eigenvectors. Furthermore, eigenvectors corresponding to distinct eigenvalues are orthogonal to each other.
3. **Diagonalization:** The matrix A can be diagonalized by a unitary transformation U such that $A = UDU^\dagger$, where D is a diagonal matrix containing the eigenvalues of A and U is a unitary matrix formed by the eigenvectors of A.

Simple Explanation:

1. **Real Eigenvalues:** For Hermitian matrices, all eigenvalues are real. This is significant because it ensures stability and specific properties in various applications, including physics and optimization.
2. **Orthogonal Eigenvectors:** Each eigenvalue has its set of corresponding eigenvectors. These sets are not only linearly independent but also orthogonal (perpendicular) to each other when corresponding to different eigenvalues. This orthogonality property is crucial in many applications, like quantum mechanics.
3. **Diagonalization:** The Spectral Theorem guarantees that a Hermitian matrix can be broken down into a form where the matrix becomes diagonal. This diagonal matrix consists of the eigenvalues of the original matrix, and the unitary matrix contains the corresponding eigenvectors. This representation simplifies many computations and analyses.

Importance:

The Spectral Theorem plays a vital role in various areas:

- **Quantum Mechanics:** It's essential for understanding observables like energy, spin, and angular momentum, which are represented by Hermitian operators.
- **Signal Processing:** It's used in areas like Fourier analysis, where Hermitian matrices represent self-adjoint linear operators.
- **Optimization:** In fields like machine learning, the Spectral Theorem is essential in understanding the geometry of optimization problems.

In summary, the Spectral Theorem for Hermitian matrices provides deep insights into the nature of these matrices, their eigenvalues, and eigenvectors, and it has profound implications across mathematics, physics, engineering, and various scientific disciplines.

[1]:
```
# Consider Below example for Spectral theorem.
import numpy as np

# Define the Hermitian matrix
A = np.array([[3, 2j],
              [-2j, 4]])
print("A=",A)
print("******************************")
# Find eigenvalues and eigenvectors
eigenvalues, eigenvectors = np.linalg.eig(A)
print("Eigenvalues:", eigenvalues)
print("******************************")
print("Eigenvectors:\n", eigenvectors)
print("******************************")
# Construct matrix P and D for diagonalization
P = eigenvectors
D = np.diag(eigenvalues)
# Check if A = PDP^(-1)
reconstructed_A = P @ D @ np.linalg.inv(P)
print("\nReconstructed A (PDP^(-1)):\n", reconstructed_A)
print("******************************")

print("*******Hence satisfied Spectral theorem*******")
```

```
A= [[ 3.+0.j  0.+2.j]
 [-0.-2.j  4.+0.j]]
******************************
Eigenvalues: [1.43844719+0.j 5.56155281+0.j]
******************************
Eigenvectors:
 [[ 0.78820544+0.j         -0.        +0.61541221j]
 [-0.        +0.61541221j  0.78820544+0.j        ]]
******************************

Reconstructed A (PDP^(-1)):
 [[3.+0.j 0.+2.j]
 [0.-2.j 4.+0.j]]
******************************
*******Hence satisfied Spectral theorem*******
```

7 Information in Single Systems :

7.1 Descriptions of Quantum Information :

Simplified Description :

Quantum States represented by **vectors**, and operations are represented by **Unitary Matrices**. Here, we will restrict our attention in to the comparatively *simple setting* in which a **single system**

is considered in isolation.

There are, in fact, **two** common mathematical descriptions of quantum information. The one introduced here is the simpler of the two. This description is **sufficient for understanding many (or perhaps most) quantum algorithms**, and is a natural place to start from a pedagogical viewpoint.

But if we are considering the noise present in information then this simple approach does not provide sufficient information about them.

General Description :

A more general, and ultimately more **powerful description** of quantum information, in which quantum states are represented by **density matrices**, are to be considered.

- The density matrix description is essential to the study of quantum information, for several reasons.
- As examples, they can be **used to model the effects of noise on quantum computations**, or **the state of one piece of an entangled pair**.
- More generally, density matrices serve as a **mathematical basis** for quantum information theory and quantum cryptography, and are quite beautiful from a mathematical perspective.

7.2 CLassical Information in Single Systems :

To describe quantum information and how it works, we will begin with an overview of classical information.

What is the requirement for Classical Information in Quantum Computing ?

- Although quantum and classical information are different in some pretty spectacular ways, their mathematical descriptions are actually quite similar.
- Classical information also serves as a familiar point of reference when studying quantum information, as well as a source of analogy that goes a surprisingly long way.
- Indeed, it is not at all unreasonable to claim that one cannot truly understand quantum information without understanding classical information.
- In addition to highlighting the aspects of classical information that are most relevant to an introduction to quantum information, this section introduces the Dirac notation, which is often used to describe vectors and matrices in quantum information and computation.
- As it turns out, the Dirac notation is not specific to quantum information: it can equally well be used in the context of classical information, as well as for many other settings in which vectors and matrices arise.

7.2.1 Classical States and Probability Vectors :

Consider that we have a system **X** which stores the information. Let's assume that this system can be in one of a finite number of classical states at each instant.

- Here **classical state** should be considered as a configuration that can be recognized and described unambiguously.

Let us use $\sum$ as the notation that denotes classical-states set. For example : 1. If X is a **Bit**, then $\sum$={0,1}. 2. If X is a **Six-sided die**, then $\sum$={1,2,3,4,5,6}. 3. If X is a **Switch of a standard electric fan**, then $\sum$={high,medium,low,off}.

In mathematical terms, the specification of the classical states of a system are, in fact, the starting point: we define a bit to be a system that has classical states **0** and **1** and likewise for systems having different classical state sets.

We naturally assume that $\sum$ as nonempty, because it is nonsensical for a physical system to have no states at all.

Although it does make sense to consider physical systems having infinitely many classical states. For the sake of convenience and brevity, we will hereafter use the term classical state set to mean any finite and nonempty set.

When thinking about X as a carrier of information, the different classical states of X could be assigned certain meanings, leading to different outcomes or consequences.

- In such cases, it may be sufficient to describe X as simply being in one of its possible classical states.
- For instance, if X is a fan switch, we might happen to know with certainty that it is set to high.

But in information processing, our knowledge of X is **uncertain**. So, we represent our knowledge of the classical state of X by assigning **probabilities** to each classical state, resulting in a ***probabilistic*** state.

For example, if X is a bit, then perhaps it is in **classical state-0** with probability of $\boxed{\frac{3}{4}}$ and in **classical state-1** with probability of $\boxed{\frac{1}{4}}$. This is called Probabilistic state of X.

$$\boxed{Pr(X=0) \ = \frac{3}{4} \ and \ Pr(X=1) \ = \frac{1}{4}}$$

The better way to represent this states is by using column vectors.

$$\begin{pmatrix} \frac{3}{4} \\ \frac{1}{4} \end{pmatrix}$$

The probability of the bit being **0** is placed at the **top** of the vector and the probability of the bit being **1** is placed at the **bottom**, simply because this is the conventional way to order the set **{0,1}**.

In general, we can represent a probabilistic state of a system having any classical state set in the same way, as a vector of probabilities. The probabilities can be ordered in any way we choose — which is typically determined by some natural or default way to order the classical state set we're working with.

To be precise, we can represent any probabilistic state through a column vector satisfying two properties:

1. All entries of the vector are **nonnegative real numbers**.
2. The sum of the entries is equal to **1**.

Conversely, any column vector that satisfies these two properties can be taken as a representation of a probabilistic state. Hereafter, we will refer to vectors of this form as ***probability vectors***.

7.2.2 Measuring Probabilistic Areas :

Now let us briefly consider what happens if we measure a system when it is in a probabilistic state. By measuring a system, we mean that we look at the system and unambiguously recognize whatever classical state it is in. Intuitively, we can never "see" a system in a probabilistic state; a measurement will yield exactly one of the allowed classical states.

Measurement changes our knowledge of the system, and therefore changes the probabilistic state that we associate with that system: if we recognize that X is in the classical state $\mathbf{a} \in \Sigma$, then the new probability vector representing our knowledge of X becomes a vector having a **1** in the entry corresponding to **a** and **0** for all other entries.

This vector indicates that X is in the classical state **a** with certainty, which we know having just recognized it.

We denote the vector just described, meaning the vector having a **1** in the entry corresponding to **a** and **0** for all other entries, by $|a\rangle$. This vector is read as ***ket-a*** for a reason that will be explained shortly. Vectors of this sort are also called ***standard basis vectors***.

For example, assuming that the system we have in mind is a bit, the standard basis vectors are given by

$$\boxed{|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}}$$

Notice that any two-dimensional column vector can be expressed as a linear combination of these two vectors. For example, we have

$$\boxed{\begin{pmatrix} \frac{3}{4} \\ \frac{1}{4} \end{pmatrix} = \frac{3}{4}|0\rangle + \frac{1}{4}|1\rangle}$$

This fact naturally generalizes to any classical state set: any column vector is a linear combination over the classical states. We will often express vectors in this way.

We measure a classical state $a \in \Sigma$ where Σ is the set of all possible states. This changes the probabilistic state of X (from our viewpoint). This means we recognized X at certain state **a**. This means $Pr(\mathsf{X} = a) = 1$. This **Probabilistic state** of a is represented as $|a\rangle$.

If we consider the above example of bit being in two states **{0,1}** when we measure the trasition takes place. Measuring X chooses any trasistion at random i.e, transition of being in $|0\rangle$ or $|1\rangle$.

Returning to the change of a probabilistic state upon being measured, we may note the following connection to our everyday experiences. Suppose we flip a fair coin, but cover up the coin before looking at it. We would then say that its probabilistic state is

$$\begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} = \frac{1}{2}|\text{heads}\rangle + \frac{1}{2}|\text{tails}\rangle$$

Here, the classical state set of our coin is heads, tails. We'll choose to order these states as *heads first, tails second.*

$$|\text{heads}\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad |\text{tails}\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

If we were to uncover the coin and look at it, we would see one of the two classical states: **heads or tails**.

Supposing that the result were tails, we would naturally update our description of the probabilistic state of the coin so that it becomes tails⟩.

Of course, if we were then to cover up the coin, and then uncover it and look at it again, the classical state would still be **tails**, which is consistent with the probabilistic state being described by the vector |tails⟩.

This may seem trivial, and in some sense it is. However, quantum systems behave in an entirely analogous way, yet their measurement properties are frequently considered unusual or "spooky".

By establishing the measurement properties of classical systems, the analogous behavior for quantum information might seem less unusual.

7.2.3 Deterministic Operations :

An operation is `deterministic` if the input completely determines the result without any element of chance or uncertainty. Here, we are interested in deterministic operations, where the output is entirely determined by the input.

When we perform an Operation on a System, its state generally changes. In this case `deterministic` refers that the result of the operation depends entirely on whatever the classical state the system was in before the operation was performed.

- This means there is no element of **`chance`** when the deterministic operation is performed. There is no concept of randomness or uncertainty involved here.

Mathematically, the deterministic operations are often represented as functions.

First, there are deterministic operations, where each classical state $\mathbf{a} \in$ is transformed into $f(a)$ for some function f of the form $\boxed{f : \Sigma \to \Sigma}$.

For example, if $\Sigma = \{0, 1\}$, there are four functions of this form, f_1, f_2, f_3, and f_4, which can be represented by tables of values as follows:

a	$f_1(a)$
0	0
1	0

a	$f_2(a)$
0	0
1	1

a	$f_3(a)$
0	1
1	0

a	$f_4(a)$
0	1
1	1

The first and last of these functions are constant: $f_1(a) = 0$ and $f_4(a) = 1$ for each $a \in \Sigma$.

The middle two are not constant, they are balanced, in the sense that the two possible output values occur the same number of times as we range over the possible inputs.

- The function f_2 is the identity function: $f_2(a) = a$ for each $a \in \Sigma$.
- And f_3 is the function $f_3(0) = 1$ and $f_3(1) = 0$, which is better-known as the `NOT` function.

The actions of deterministic operations on probabilistic states can be represented by `matrix-vector multiplication`. Specifically, the matrix $\mathbf{M}$ that represents a given function $f : \Sigma \to \Sigma$ is the one that satisfies

$$\boxed{M|a\rangle = |f(a)\rangle} \textit{ for every } a \in \Sigma$$

Such a matrix always exists and is unique.

For example, the matrices $M_1, \ldots, M_4$ corresponding to the functions $f_1, \ldots, f_4$ above are as follows:

$$M_1 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad M_3 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad M_4 = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$$

Matrices that represent deterministic operations always have exactly one 1 in each column, and 0 for all other entries.

The matrices M_1 to M_4 are constructed to capture the transformation of states based on each function. These matrices follow a specific pattern: a single "1" in each column and 0 elsewhere.

A convenient way to represent matrices of these and other forms makes use of an analogous notation for row vectors to the one for column vectors discussed previously.

We denote by $\langle a|$ the row vector having a 1 in the entry corresponding to a and **zero** for all other entries, for each $a \in \Sigma$. This vector is read as "bra a". For example, if $\Sigma = \{0, 1\}$, then

$$\langle 0| = \begin{pmatrix} 1 & 0 \end{pmatrix} \quad \text{and} \quad \langle 1| = \begin{pmatrix} 0 & 1 \end{pmatrix}$$

For an arbitrary choice of a classical state set Σ, viewing row vectors and column vectors as matrices and performing the matrix multiplication $|b\rangle\langle a|$, one obtains a square matrix having a 1 in the entry corresponding to the pair (b, a), meaning that the row of the entry corresponds to b and the column corresponds to a, and 0 for all other entries. For example,

$$|0\rangle\langle 1| = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

Using this notation, for any function $f : \Sigma \to \Sigma$, we may express the matrix M corresponding to the function f as

$$\boxed{M = \sum_{a \in \Sigma} |f(a)\rangle\langle a|}$$

Now, if we again think about vectors as matrices, but this time consider the multiplication $\langle a||b\rangle$, then we obtain a 1×1matrix, which we can think about as a **`scalar`** (i.e., a number).

For the sake of tidiness, we write this product as $\langle a|b\rangle$ rather than $\langle a||b\rangle$. This product satisfies the following simple formula:

$$\boxed{\langle a|b\rangle = \begin{cases} 1 & a = b \\ 0 & a \neq b. \end{cases}}$$

Using this observation, together with the fact that matrix multiplication is `associative and linear`, we obtain

$$\boxed{M|b\rangle = \left(\sum_{a\in\Sigma} |f(a)\rangle\langle a|\right)|b\rangle = \sum_{a\in\Sigma} |f(a)\rangle\langle a|b\rangle = |f(b)\rangle}$$

for each $b \in \Sigma$, which is what we require of M. Thus, the above example indicates the transformed state after applying the identity function.

7.2.4 Probabilistic Operations :

In addition to deterministic operations, we have probabilistic operations. This means the classical operations can introduce the **`randomness`** or **`uncertainty`**.

For example, consider an operation on a bit where,

- if the classical state of the bit is 0, it is left alone, (***do nothing***)
- and if the classical state of the bit is 1, it is flipped to 0 with probability 1/2.

This operation is represented by the matrix

$$\begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}$$

For an arbitrary choice of a classical state set, we can describe the set of all probabilistic operations in mathematical terms as those that are represented by **`stochastic matrices`**, which are matrices satisfying these two properties:

1. All entries are nonnegative real numbers.
2. The entries in every column sum to 1.

Equivalently, stochastic matrices are matrices whose columns all form probability vectors.

We can think about probabilistic operations at an intuitive level as ones where randomness might somehow be used or introduced during the operation, just like in the example above.

With respect to the stochastic matrix description of a probabilistic operation, each column can be viewed as a vector representation of the probabilistic state that is generated given whatever classical state input corresponds to that column. This means *`Stochastic matrices`* represent operations where each **column is like a vector** showing the ***probabilities of different outcomes given the corresponding classical state***.

We can also think about stochastic matrices as being exactly those matrices that always map probability vectors (vectors representing probabilities of different outcomes) to probability vectors.

- That is, stochastic matrices always map probability vectors to probability vectors, and any matrix that always maps probability vectors to probability vectors must be a stochastic matrix.

Finally, another way to think about probabilistic operations is that they are **random choices of deterministic operations**. For instance, we can think about the operation in the example above as applying either the **`identity function`** or the **`constant 0 function`**, each with probability 1/2. This is consistent with the equation

$$\begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}.$$

The matrix representing a probabilistic operation can be expressed as a combination of deterministic operations weighted by their respective probabilities.

Such an expression is always possible, for an arbitrary choice of a classical state set and any stochastic matrix with rows and columns identified with that classical state set.

7.2.5 Compositions of probabilistic operations :

Suppose that X is a system having classical state set Σ, and $M_1, \ldots, M_n$ are stochastic matrices representing probabilistic operations on the system X.

If the first operation M_1 is applied to the probabilistic state represented by a probability vector u, the resulting probabilistic state is represented by the vector $M_1 u$.

If we then apply the second probabilistic operation M_2 to this new probability vector, we obtain the probability vector

$$\boxed{M_2(M_1 u) = (M_2 M_1)u}$$

The equality follows from the fact that matrix multiplication (which includes matrix-vector multiplication as a special case) is an **`associative operation`**. Thus, the probabilistic operation obtained by composing the first and second probabilistic operations, where we first apply M_1 and then apply M_2, is represented by the matrix $M_2 M_1$, which is necessarily stochastic.

More generally, composing the probabilistic operations represented by the matrices $M_1, \ldots, M_n$ in that order, meaning that M_1 is applied first, M_2 is applied second, and so on, with M_n applied last, is represented by the matrix

$$M_n, \cdots, M_1$$

Note that the ordering is important here: *although matrix multiplication is associative, it is not a commutative operation in general.*

For example, if we have

$$M_1 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad M_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

then,

$$M_2 M_1 = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \quad \text{and} \quad M_1 M_2 = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$$

That is, the *order in which probabilistic operations are composed matters*: changing the order in which operations are applied in a composition can change the resulting operation.

7.3 Quantum Information in Single Systems :

Now we move on to quantum information, where we make a different choice for the type of vector that represents a state — in this case, a quantum state — of the system being considered.

Quantum information works in a very similar way to classical information but, with some key differences.

7.3.1 Quantum State Vectors :

The major difference we can see between classical and Quantum information is regarding **`how we define a state`**. And the procedure to define the Quantum State largely determines how Quantum Information works.

A **`Quantum State`** of a system is represented by a *column vector* whose indices are placed in correspondence with the classical states of that system.

This means The quantum state of a system is represented by a column vector, similar to ***`probabilistic states`***. As before, the indices of the vector, label the classical states of the system.

Vectors representing quantum states are characterized by these two properties:

1. The entries of a quantum state vector are **`complex numbers`**.
2. The **`sum of the absolute values squared of the entries`** of a quantum state vector is 1.

Thus, in contrast to the probabilistic case, vectors representing quantum states need not have nonnegative real number entries, and it is the sum of the absolute values squared of the entries (as opposed to the sum of the entries) that must equal 1.

Simple as these changes are, they give rise to all the differences between quantum and classical information. Any speedup from a quantum computer, or improvement in a communication protocol, ultimately derives from these simple mathematical changes.

The **`Euclidean norm`** of a column vector defined as follows:

$$v = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} \implies \|v\| = \sqrt{\sum_{k=1}^{n} |\alpha_k|^2}$$

The condition that the sum of the absolute values squared of a quantum state vector equals 1 is therefore equivalent to that vector having Euclidean norm equal to 1. That is, quantum state vectors are unit vectors with respect to the Euclidean norm.

Examples of qubit states

The term `qubit` refers to a quantum system whose classical state set is {0,1}. That is, a qubit is really just a bit — but by using this name we explicitly recognize that this bit can be in a quantum state.

These are examples of quantum states of a qubit:

$$\boxed{\begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle \quad \text{and} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle}$$

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle, \tag{1}$$

and

$$\begin{pmatrix} \frac{1+2i}{3} \\ -\frac{2}{3} \end{pmatrix} = \frac{1+2i}{3}|0\rangle - \frac{2}{3}|1\rangle$$

The first two examples, $|0\rangle$ and $|1\rangle$ illustrate that standard basis elements are valid quantum state vectors: their entries are `complex numbers` (where the imaginary part of these numbers happens to be 0) and computing the sum of the absolute values squared of the entries yields

$$|1|^2 + |0|^2 = 1 \quad \text{and} \quad |0|^2 + |1|^2 = 1,$$

as required.

Similar to the classical setting, we associate the quantum state vectors $|0\rangle$ and $|1\rangle$ with a qubit being in the classical state 0 or 1, respectively, with certainty.

For the other two examples, we again have `complex number` entries, and computing the sum of the absolute value squared of the entries yields

$$\boxed{\Longrightarrow \left|\frac{1}{\sqrt{2}}\right|^2 + \left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2} + \frac{1}{2} = 1}$$

and

$$\boxed{\Longrightarrow \left|\frac{1+2i}{3}\right|^2 + \left|-\frac{2}{3}\right|^2 = \frac{5}{9} + \frac{4}{9} = 1}$$

These are therefore valid quantum state vectors.

Note that they are `linear combinations` of the states $|0\rangle$ and $|1\rangle$. We often say that these states are **`superpositions of the 0 and 1 states`**. Within the context of quantum states, `superposition` and `linear combination` are essentially synonymous.

The example of a qubit state vector above is very commonly encountered — it is called the ***`plus state`*** and is denoted as follows:

$$\boxed{|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle} \qquad \text{(plus state)}$$

We also use the notation

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

to refer to a related quantum state vector where the second entry is negative rather than positive, and we call this state the ***minus state***.

This sort of notation, where some symbol other than one referring to a classical state appears inside of a ket, is common — we can use whatever name we wish inside of a ket to name a vector. Indeed, it is quite common to use the notation $|\Psi\rangle$, or other names in place of Ψ, to refer to an arbitrary vector that may not necessarily be a standard basis vector.

Notice that if we have a vector $|\Psi\rangle$ whose indices correspond to some classical state set Σ, and if $a \in \Sigma$ is an element of this classical state set, then the (matrix) product $\langle a||\Psi\rangle$ *is equal to the entry of the vector* $|\Psi\rangle$ whose index corresponds to a.

This means The entry of the vector $|\psi\rangle$ corresponding to the classical state a is the specific value in the vector at the position indexed by a. For example, If

$$\boxed{\Sigma = \{0,1\}, \textit{and } |\Psi\rangle = \begin{pmatrix} x_0 \\ x_1 \end{pmatrix} \implies \langle 0||\Psi\rangle = x_0 \textit{ and } \langle 1||\Psi\rangle = x_1}$$

In simpler terms, the expression $\langle a||\Psi\rangle$ retrieves the specific value from the vector $|\Psi\rangle$ corresponding to the classical state a. This operation is fundamental in quantum mechanics and linear algebra, where it helps extract information about specific components of quantum states.

As we did when $|\Psi\rangle$ is a `standard basis vector`, we write $\langle a|\Psi\rangle$ rather than $\langle a||\Psi\rangle$ for the sake of readability.

Another example, if $\Sigma = \{0,1\}$ and $|\Psi\rangle = \frac{1+2i}{3}|0\rangle - \frac{2}{3}|1\rangle$ then,

$$\boxed{|\Psi\rangle = \frac{1+2i}{3}|0\rangle - \frac{2}{3}|1\rangle = \begin{pmatrix} \frac{1+2i}{3} \\ -\frac{2}{3} \end{pmatrix} \implies \langle 0|\Psi\rangle = \frac{1+2i}{3} \quad \text{and} \quad \langle 1|\Psi\rangle = -\frac{2}{3}} \tag{2}$$

It must be understood when using this notation that $\langle\Psi|$ refers to the row vector obtained by taking the `conjugate-transpose` of the column vector $|\Psi\rangle$, where the vector is transposed from a column vector to a row vector and each entry is replaced by its complex conjugate.

For example, if $|\Psi\rangle$ is the vector defined in (2), then

$$\boxed{\langle\Psi| = \frac{1-2i}{3}\langle 0| - \frac{2}{3}\langle 1| = \begin{pmatrix} \frac{1-2i}{3} & -\frac{2}{3} \end{pmatrix}}$$

Quantum states of other systems :

We can consider quantum states of systems having arbitrary classical state sets.

For example, here is a quantum state vector for an `electrical fan switch`:

$$\begin{pmatrix} \frac{1}{2} \\ ]0 \\ ]-\frac{i}{2} \\ ]\frac{1}{\sqrt{2}} \end{pmatrix} = \frac{1}{2}|\text{high}\rangle - \frac{i}{2}|\text{low}\rangle + \frac{1}{\sqrt{2}}|\text{off}\rangle$$

- The assumption here is that the classical states are ordered as `high`, `medium`, `low`, and `off`.
- There may be no particular reason why one would want to consider the quantum state of an electrical fan switch, but it is possible in principle.

Here's another example, this time of a `quantum decimal digit` whose classical states are $0, 1, \ldots, 9$:

$$\frac{1}{\sqrt{385}} \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \end{pmatrix} = \frac{1}{\sqrt{385}} \sum_{k=0}^{9} (k+1)|k\rangle$$

- This example illustrates the convenience of writing state vectors using the Dirac notation.
- For this particular example, the column vector representation is merely cumbersome — but if there were significantly more classical states it would become unusable.

The Dirac notation, in contrast, supports precise descriptions of large and complicated vectors in a compact form.

- The Dirac notation also allows for the expression of vectors where different aspects of the vectors are `indeterminate` (meaning unknown or not yet established).

For example, for an arbitrary classical state set Σ, we can consider the quantum state vector

$$\frac{1}{\sqrt{|\Sigma|}} \sum_{a \in \Sigma} |a\rangle,$$

- which is a uniform superposition over the classical states in Σ. (Here, the notation $|\Sigma|$ refers to the number of elements in Σ.)

We'll encounter much more complicated expressions of quantum state vectors, where the use of column vectors would be impractical or impossible. In fact, we'll mostly abandon the column vector representation of state vectors, except for vectors having a `small number of entries` (often in the context of examples), where it may be helpful to display and examine the entries explicitly.

There is one more reason why expressing state vectors using the Dirac notation is generally more convenient: *it alleviates the need to explicitly specify an ordering of the classical states (or, equivalently, the correspondence between classical states and vector indices).*

For example, a quantum state vector for a system having `classical state set` $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$, such as

$$\frac{1}{2}|\clubsuit\rangle + \frac{i}{2}|\diamondsuit\rangle - \frac{1}{2}|\heartsuit\rangle - \frac{i}{2}|\spadesuit\rangle,$$

- is unambiguously described by this expression, and there really is no need to choose or specify an ordering of this classical state set in order to make sense of the expression.
- In this case, it is not difficult to simply specify an ordering of the standard card suits — for instance, we could order them as follows: $\clubsuit$, $\diamondsuit$, $\heartsuit$, $\spadesuit$.
- If we choose this particular ordering, the quantum state vector above would be represented by the column vector

$$\begin{pmatrix} \frac{1}{2} \\ \frac{i}{2} \\ -\frac{1}{2} \\ -\frac{i}{2} \end{pmatrix}$$

In general, however, it is convenient to simply ignore the question of how classical state sets are ordered, and take the view that quantum state vectors are directly indexed by classical states.

7.3.2 Measuring Quantum States :

Next let us consider what happens when a quantum state is measured, focusing on a simple type of measurement known as a ***`standard basis measurement`***.

Similar to the probabilistic setting, when a system in a quantum state is measured, the observer performing the measurement won't see a quantum state vector, but rather some classical state. In this sense, ***measurements act as the interface between quantum and classical information, through which classical information is extracted from quantum states***.

The rule is simple: *if a quantum state is measured, each classical state of the system results with probability equal to the absolute value squared of the entry in the quantum state vector corresponding to that classical state.* This is known as the **`Born rule`** in quantum mechanics.

Notice that this rule is consistent with the requirement that the **absolute values squared of the entries in a quantum state vector sum to 1**, as it implies that the *probabilities of different classical state measurement outcomes sum to 1.*

For example, measuring the plus state

$$\boxed{|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle}$$

results in the two possible outcomes, 0 and 1 with probabilities as follows.

$$\implies Pr(outcome\ is\ 0) = \left|\langle 0|+\rangle\right|^2 = \left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$$

$$\implies Pr(outcome\ is\ 1) = \left|\langle 1|+\rangle\right|^2 = \left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$$

Similarly, for the minus state

$$\boxed{|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle}$$

results in exactly the same probabilities for the two outcomes.

$$\implies Pr(outcome\ is\ 0) = \left|\langle 0|-\rangle\right|^2 = \left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$$

$$\implies Pr(outcome\ is\ 1) = \left|\langle 1|-\rangle\right|^2 = \left|\frac{-1}{\sqrt{2}}\right|^2 = \frac{1}{2}$$

This suggests that, as far as standard basis measurements are concerned, the `plus` and `minus` states are exactly the same.

Of course,

- measuring the quantum state $|\mathbf{0}\rangle$ results in the classical state **0** with certainty,
- and likewise measuring the quantum state $|\mathbf{1}\rangle$ results in the classical state **1** with certainty.

This is consistent with the identification of these quantum states with the system being in the corresponding classical state.

As a final example, measuring the state

$$\boxed{|\Psi\rangle = \begin{pmatrix} \frac{1+2i}{3} \\ -\frac{2}{3} \end{pmatrix} = \frac{1+2i}{3}|0\rangle - \frac{2}{3}|1\rangle}$$

has probabilities given by

$$\implies Pr(outcome\ is\ 0) = \left|\langle 0|\Psi\rangle\right|^2 = \left|\frac{1+2i}{3}\right|^2 = \frac{5}{9}$$

$$\implies Pr(outcome\ is\ 1) = \left|\langle 1|\Psi\rangle\right|^2 = \left|-\frac{2}{3}\right|^2 = \frac{4}{9}$$

7.3.3 Unitary Operations :

A unitary operation is a fundamental concept in linear algebra and quantum mechanics. In quantum computing, unitary operations play a central role in manipulating quantum states. A unitary operation on a quantum system preserves the norm of the state vectors and is reversible.

Thus far, it may not be evident why quantum information is fundamentally different from classical information. That is, when a quantum state is measured, the **probability to obtain each classical state** is given by the **absolute value squared of the corresponding vector entry** — ***so why not simply record these probabilities in a probability vector?***

The answer, at least in part, is that the `set of allowable operations` that can be performed on a *quantum state* is different than it is for classical information.

Similar to the probabilistic setting, operations on quantum states are **linear mappings** — but rather than being represented by ***stochastic matrices*** as in the classical case, operations on quantum state vectors are represented by ***unitary matrices***.

A square matrix U having complex number entries is **unitary** if it satisfies the equations

$$\boxed{\mathsf{U} \cdot \mathsf{U}^\dagger = \mathbb{1} \ = \mathsf{U}^\dagger \cdot \mathsf{U}} \tag{3}$$

- Here $\mathbb{1}$ represents Identity matrix
- $\mathsf{U}^\dagger$ is complex conjugate of U meaning the matrix obtained by transposing U and taking the complex conjugate of each entry.

$$\mathsf{U}^\dagger \ = \overline{\mathsf{U}^T}$$

If either of the two equalities numbered (3) above is true, then the other must also be true. Both equalities are equivalent to $\mathsf{U}^\dagger$ being the inverse of U :

$$\mathsf{U}^{-1} \ = \mathsf{U}^\dagger$$

If M is not a **square matrix** then it could be $\mathsf{M}^\dagger \cdot \mathsf{M} \ = \mathbb{1}$ and $\mathsf{M}^\dagger \cdot \mathsf{M} \neq \mathbb{1}$, for instance. The equivalence of the two equalities in the third equation above is only `true for square matrices`.

The condition that U is **unitary** `is equivalent to` the condition that multiplication by U `does not change` the **Euclidean norm** of any vector.

- This means any matrix of $n \times n$ is **UNITARY** if and only if

$$\boxed{\left\| \mathsf{U}_{n\times n} \cdot |\Psi\rangle \right\| = \left\| |\Psi\rangle \right\|}$$

  for every n dimensional column vector $|\Psi\rangle$ with complex number entries.

- Thus, because the set of all quantum state vectors is the same as the set of vectors having Euclidean norm equal to **1**, ***multiplying a unitary matrix to a quantum state vector results in another quantum state vector***. So, here $|\Psi\rangle$ is also a **Quantum State vector**.
- `Unitary matrices` are exactly the set of linear mappings which always transform quantum state vectors to other quantum state vectors.
- Here is a resemblance to the **classical probabilistic** case, where operations are associated with `stochastic matrices`, which are the ones that ***always transform probability vectors into probability vectors***.
- ***A square matrix is unitary if and only if it doesn't change the Euclidean norm of any vector when you multiply.***

Key Characteristics of Unitary Operations:

1. **Preservation of Norm:** A unitary operation preserves the normalization of quantum states. That is, the operation doesn't change the length (or norm) of the quantum state vectors. Mathematically, for a quantum state vector $|\psi\rangle$, after applying a unitary operation U, the norm remains unchanged: $\|U|\psi\rangle\| = \||\psi\rangle\|$.

2. **Reversibility:** Unitary operations are reversible transformations. This means that for every unitary operation applied to a quantum state, there exists an inverse operation that can revert the state back to its original form.

3. **Compositions of Unitary Operations:** When multiple unitary operations are applied sequentially (in a sequence known as a quantum circuit), the resulting transformation remains unitary. This property is crucial in quantum algorithms, where sequences of unitary gates represent computations.

Representation in Quantum Computing:

In the context of quantum computing, quantum gates represent unitary operations. These gates are applied to qubits (quantum bits) to perform various operations on quantum states. Examples of common unitary gates include:

- Hadamard gate
- Pauli gates (X, Y, Z)
- Controlled gates (CNOT, controlled-phase, etc.)
- Phase gates (S, T)

In summary, unitary operations are essential in quantum computing as they provide reversible and norm-preserving transformations, enabling the manipulation and evolution of quantum states in a well-defined and reversible manner.

Example: Hadamard Gate as a Unitary Operation

The Hadamard gate **H** is represented by the following matrix:

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

This gate performs a transformation on a single qubit, and it's often used in quantum algorithms, including quantum teleportation and the creation of superposition states.

Application of Hadamard Gate:

Suppose we have a single qubit state initially in the basis state$|0\rangle$ also denoted as

$$\implies |\psi\rangle = |0\rangle$$

- Applying the Hadamard gate **(H)** to this qubit state $|0\rangle$ yields:

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

This represents the state $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$, which is a superposition state known as the equal superposition of $|0\rangle$ and $|1\rangle$.

Properties of Hadamard Gate:

1. **Norm Preservation:** The Hadamard gate is unitary, preserving the normalization of the state. The squared magnitudes of coefficients sum up to 1, maintaining the norm.

2. **Reversibility:** Applying **H** gate twice (i.e., **H.H = I**) returns the qubit back to its original state, showing the reversibility property of unitary operations.

The Hadamard gate exemplifies a unitary operation that transforms a quantum state by creating superposition, a crucial concept in quantum computing. It maintains the norm of the state vector and is reversible, meeting the criteria of a unitary operation.

So these are the operators which decide how a state develops with time (**from Postulate-5**) when you are not making a measurement.

The process of measurement makes it a discontinuous process, makes the state collapse to one of the eigen states or classical states of the system, and so therefore, the quantum computing deals with two aspects of this computation process. For example One is, it means that when it is passing through transistors and gates.

Here we are not trying to find out intermediate results, the system simply follows a unitary operation, but the moment we decide we want to do a Read operation to find out what is the result, we must make a measurement and when we make a measurement we can at best get a probabilistic answer. And the result that we will get will be one of the permitted results, but we have no way of telling which result we will get.

Let's use an analogy with a coin to explain the concept of a unitary operation, particularly the **Hadamard gate** in quantum computing.

Analogy: Coin Flipping with a Hadamard-like Operation

Imagine a classical coin that can be in one of two states: **heads (H) or tails (T)**. Initially, the coin is in a known state, say heads (H).

1. **Starting State:**
   - We represent the initial state of the coin as $|H\rangle$.
2. **Hadamard-like Operation:**
   - Now, let's introduce an operation similar to a Hadamard gate, denoted as **C**.
   - This **C** operation represents a magical coin flipper, which, when applied to the state $|H\rangle$, transforms the coin's state using a particular rule:

$$C(|H\rangle) = \frac{|H\rangle + |T\rangle}{\sqrt{2}}$$

3. **Interpretation:**
   - Applying this **C** operation to the initial state $|H\rangle$ results in a superposition of states:

$$\frac{|H\rangle + |T\rangle}{\sqrt{2}}$$

     .
   - This represents a magical coin state where, due to the operation, the coin is both in a `state of heads and tails simultaneously`.
4. **Properties:**

- **Norm Preservation:** Just like in quantum computing, this magical operation preserves the norm, ensuring that the probabilities of the coin being in either heads or tails add up to 1.
- **Reversibility:** Applying the same operation again ($\mathbf{C}.\mathbf{C} = \mathbf{I}$) would bring the coin back to its initial state of heads.

In this analogy, the **C** operation serves as an analogy to the Hadamard gate in quantum computing.

- It transforms the state of a classical coin, creating a superposition where the coin appears to be in both heads and tails simultaneously.

Important examples of unitary operations on qubits : The following list describes some important unitary operations on qubits.

1. Pauli Operations :

The four Pauli matrices are as follows:

$$\boxed{\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}}$$

Where, $\mathbb{1}$ refers to identity matrix.

A common notation, which we will often use, is $\mathsf{X} = \sigma_x$, $\mathsf{Y} = \sigma_y$, and $\mathsf{Z} = \sigma_z$— but we need to be aware that the letters $\mathsf{X}, \mathsf{Y}, \mathsf{Z}$ are also commonly used for other purposes.

- For suppose, The X or σ_x operation is also called a **bit flip** or a **NOT** operation because it induces this action on bits:
$$\boxed{\mathsf{X}|0\rangle = |1\rangle \text{ and } \mathsf{X}|1\rangle = |0\rangle}$$
- Similarly, the Z or σ_z operation is also called a **Phase Flip** because of the following action :
$$\boxed{\mathsf{X}|0\rangle = |0\rangle \text{ and } \mathsf{X}|1\rangle = -|1\rangle}$$

2. Hadamard Operations :

The Hadamard operation is described by this matrix:

$$\boxed{\mathsf{H} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}}$$

3. Phase Operation :

A phase operation is one described by the matrix:

$$\boxed{\mathsf{P}_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}}$$

For any choice of a real number θ. The operations are:

$$\boxed{\mathsf{S} = \mathsf{P}_{\frac{\pi}{2}} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \text{ and } \mathsf{T} = \mathsf{P}_{\frac{\pi}{4}} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix}}$$

are particularly important examples. Other examples include $\mathbb{1} = \mathsf{P}_0$ and $\mathsf{Z} = \mathsf{P}_{\pi}$.

These matrices or gates are also called as **Clifford** gates.

All of the matrices just defined are **unitary**, and therefore `represent quantum operations on a single qubit`.

For example, here is a calculation that verifies that H is unitary:

$$\boxed{\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}^{\dagger} \cdot \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{2}+\frac{1}{2} & \frac{1}{2}-\frac{1}{2} \\ \frac{1}{2}-\frac{1}{2} & \frac{1}{2}+\frac{1}{2} \end{pmatrix} \implies \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}$$

Here's the action of the Hadamard operation on a few qubit state vectors.

$$\mathsf{H}|0\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = |+\rangle \tag{a}$$

$$\mathsf{H}|1\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = |-\rangle \tag{b}$$

$$\mathsf{H}|+\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle \tag{c}$$

$$\mathsf{H}|-\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle \tag{d}$$

$$\boxed{\mathsf{H}\left(\frac{1+2i}{3}|0\rangle - \frac{2}{3}|1\rangle\right) = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \times \begin{pmatrix} \frac{1+2i}{3} \\ -\frac{2}{3} \end{pmatrix} \implies \begin{pmatrix} \frac{-1+2i}{3\sqrt{2}} \\ \frac{3+2i}{3\sqrt{2}} \end{pmatrix} = \frac{-1+2i}{3\sqrt{2}} \cdot |0\rangle + \frac{3+2i}{3\sqrt{2}} \cdot |1\rangle}$$

The examples (a),(b),(c), and (d) are the important transformations of Hadamard Gates.

Consider a situation in which a qubit is prepared in one of the two quantum states $|+\rangle$ and $|-\rangle$, but where it is not known to us which one it is. Measuring either state produces the same output distribution as the other: **0** and **1** both appear with equal probability $\frac{1}{2}$.

So, doing this provides no information about which of the two states $|+\rangle$ and $|-\rangle$ was originally prepared. However, if we apply a Hadamard operation and then measure,

- we obtain the outcome **0** with certainty if the original state was $|+\rangle$
- and we obtain the outcome 1, again with certainty, if the original state was $|-\rangle$.

Thus, the quantum states $|+\rangle$ and $|-\rangle$ can be discriminated perfectly. This reveals that sign changes, or more generally changes to the **phases** (which are also traditionally called *arguments*) **of the complex number entries of a quantum state vector, can significantly change that state**.

Here's another example, this time of the action of a T operation on a plus state.

$$\begin{aligned}\mathsf{T}|+\rangle &= \mathsf{T}\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right)\\ &= \frac{1}{\sqrt{2}}\mathsf{T}|0\rangle + \frac{1}{\sqrt{2}}\mathsf{T}|1\rangle\\ &= \frac{1}{\sqrt{2}}|0\rangle + \frac{1+i}{2}|1\rangle\end{aligned}$$

Notice here that we did not bother to convert to the equivalent matrix/vector forms, and instead used the linearity of matrix multiplication together with the formulas

$$\mathsf{T}|0\rangle = |0\rangle \quad \text{and} \quad \mathsf{T}|1\rangle = \frac{1+i}{2}|1\rangle$$

Where T is a **clifford gate** which is also represented as $\mathsf{P}_{\frac{\pi}{4}}$.

Along similar lines, we may compute the result of applying a Hadamard operation to the quantum state vector just obtained $\mathsf{T}|+\rangle$:

$$\begin{aligned}\mathsf{H}\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1+i}{2}|1\rangle\right) &= \frac{1}{\sqrt{2}}\mathsf{H}|0\rangle + \frac{1+i}{2}\mathsf{H}|1\rangle\\ &= \frac{1}{\sqrt{2}}|+\rangle + \frac{1+i}{2}|-\rangle\\ &= \left(\frac{1}{2}|0\rangle + \frac{1}{2}|0\rangle\right) + \left(\frac{1+i}{2\sqrt{2}}|0\rangle - \frac{1+i}{2\sqrt{2}}|0\rangle\right)\\ &= \left(\frac{1}{2} + \frac{1+i}{2\sqrt{2}}\right)|0\rangle + \left(\frac{1}{2} - \frac{1+i}{2\sqrt{2}}\right)|1\rangle\end{aligned}$$

The two approaches —

- one where we explicitly convert to matrix representations
- and the other where we use linearity and plug in the actions of an operation on standard basis states — are equivalent.

We can use whichever one is more convenient in the case at hand.

7.3.4 Compositions of Unitary Operations :

Compositions of unitary operations are represented by matrix multiplication, just like we had in the probabilistic setting. For example, if we first apply a Hadamard operation, followed by the S operation, followed by another Hadamard operation, the resulting operation (which we'll name R) is as follows:

$$\mathsf{R} = \mathsf{HSH} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1+i}{2} & \frac{1-i}{2} \\ \frac{1-i}{2} & \frac{1+i}{2} \end{pmatrix}$$

This unitary operation R is an interesting example. By applying this operation twice, which is equivalent to **squaring** its matrix representation, we obtain a ***NOT*** operation:

$$\boxed{\implies \mathsf{R}^2 = \begin{pmatrix} \frac{1+i}{2} & \frac{1-i}{2} \\ \frac{1-i}{2} & \frac{1+i}{2} \end{pmatrix}^2 = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} = \mathsf{S}}$$

That is, R is a **square root of NOT operation**. Such a behavior, where the same operation is applied twice to yield a NOT operation, is not possible for a classical operation on a single bit. This is related to the fact that you ***can't take the square root of a negative number*** and stay on the real line.

7.3.5 Qiskit Examples :

Qiskit uses the Python programming language, so before discussing Qiskit specifically it may be helpful to very briefly discuss matrix and vector computations in Python. In Python, matrix and vector computations can be performed using the array class from the **NumPy** library (which includes many additional components for numerical computation).

Here is an example of a code cell that defines two vectors, ket0 and ket1 , corresponding to the qubit state vectors $|0\rangle$ and $|1\rangle$, and displays their average.

[2]:
```
from numpy import array

ket0 = array([1, 0])
ket1 = array([0, 1])

display(ket0 / 2 + ket1 / 2)
```

```
array([0.5, 0.5])
```

[3]:
```
#We can also use "array" to create matrices that represent operations.
M1 = array([[1, 1], [0, 0]])
M2 = array([[1, 1], [1, 0]])

M1 / 2 + M2 / 2
```

[3]:
```
array([[1. , 1. ],
       [0.5, 0. ]])
```

[4]:
```
#Matrix multiplication (including matrix-vector multiplication as a special␣
 ↪case)
#can be performed using the "matmul" function from NumPy :

from numpy import matmul
```

```
display(matmul(M1, ket1))
display(matmul(M1, M2))
display(matmul(M2, M1))
```

```
array([1, 0])
```

```
array([[2, 1],
       [0, 0]])
```

```
array([[1, 1],
       [1, 1]])
```

7.4 Examples on states, measurements and Operators :

7.4.1 Defining and displaying state vectors:

Qiskit's Statevector class provides functionality for defining and manipulating quantum state vectors. The following code cell imports the Statevector class and defines a few vectors using it. (Note that we need the sqrt function from the NumPy library to compute the square roots for the vector u .)

[5]:
```
from qiskit.quantum_info import Statevector
from numpy import sqrt

u = Statevector([1 / sqrt(2), 1 / sqrt(2)])
v = Statevector([(1 + 2.0j) / 3, -2 / 3])
w = Statevector([1 / 3, 2 / 3])

print("State vectors u, v, and w have been defined.")
```

```
State vectors u, v, and w have been defined.
```

The Statevector class provides a **draw** method for displaying state vectors, including latex and text options for different visualizations, as this code cell demonstrates:

[6]:
```
display(u.draw("latex"))
display(v.draw("text"))
```

$$\frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}}{2}|1\rangle$$

```
[ 0.33333333+0.66666667j,-0.66666667+0.j        ]
```

The Statevector class also includes the **is_valid** method, which checks to see *if a given vector is a valid quantum state vector* (i.e., that it has Euclidean norm equal to 1):

[7]:
```
display(u.is_valid())
display(w.is_valid())
```

```
True
```

```
False
```

Next we will see one way that measurements of quantum states can be simulated in Qiskit, using the **measure** method from the *Statevector class*.

First, we create a ***qubit state vector v*** and then display it.

[8]:
```
v = Statevector([(1 + 2.0j) / 3, -2 / 3])
v.draw("latex")
```

[8]:

$$(\frac{1}{3} + \frac{2i}{3})|0\rangle - \frac{2}{3}|1\rangle$$

Next, running the measure method simulates a **standard basis measurement**. It returns the result of that measurement, plus the new quantum state of our system after that measurement.

[9]:
```
v.measure()
```

[9]:
```
('1',
 Statevector([ 0.+0.j, -1.+0.j],
             dims=(2,)))
```

Measurement outcomes are probabilistic, so the same method can return different results. Try running this cell a few times to see this.

As an aside, Statevector will throw an error if the measure method is applied to an invalid quantum state vector.

Statevector also comes with a sample_counts method that allows for the simulation of any number of measurements on the system.

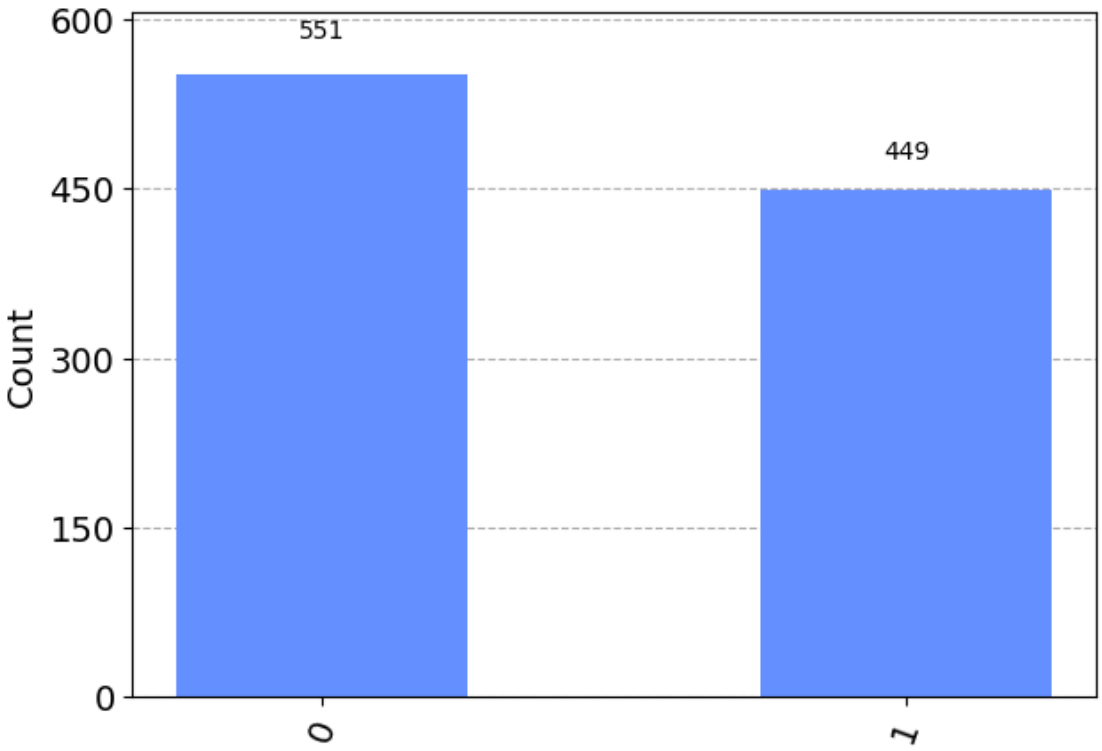
For example, the following cell shows the outcome of measuring the vector v **1000** times, which (with high probability) results in the outcome **0** approximately about **541** of the **1000** trials and the the outcome **1** approximately about **459** out of the **1000** trials).

- The cell also demonstrates the plot_histogram function for visualizing the results.

[10]:
```
from qiskit.visualization import plot_histogram

statistics = v.sample_counts(1000)
display(statistics)
plot_histogram(statistics)
```

```
{'0': 551, '1': 449}
```

[10]:

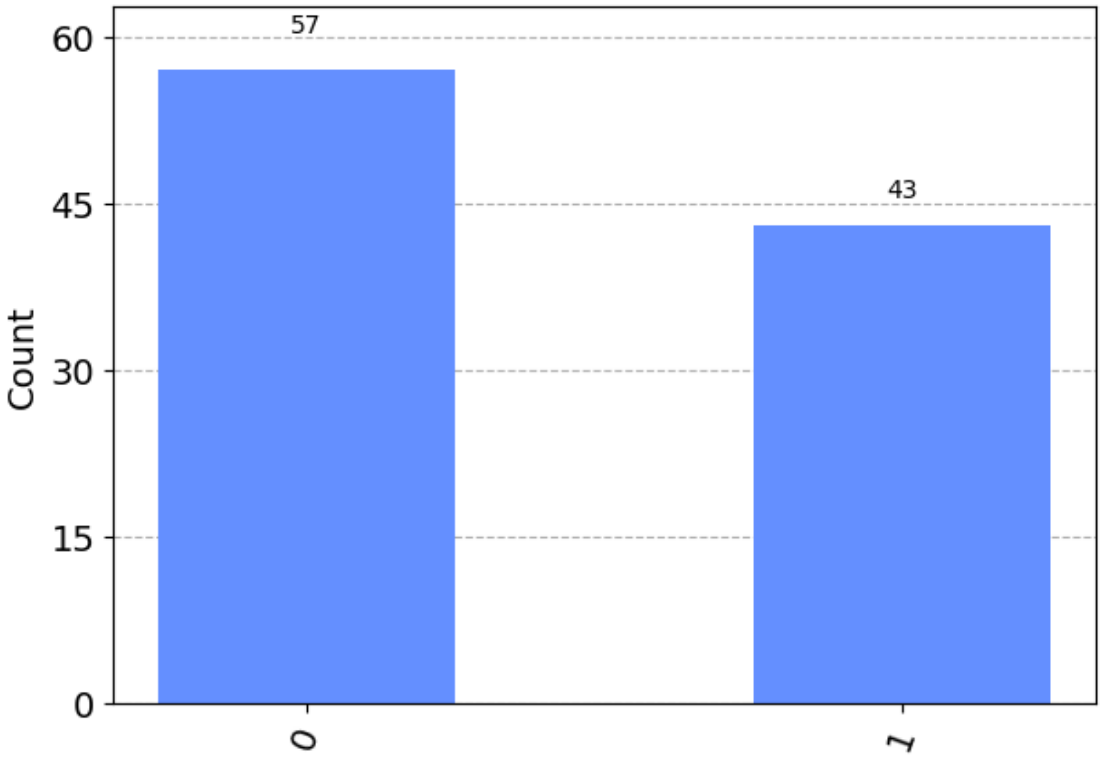


Running the cell multiple times and trying different numbers of samples in place of **1000** may be helpful for developing some intuition for how the number of trials influences the estimated probabilities.

[11]:
```
#if we consider only 100 samples
statistics = v.sample_counts(100)
display(statistics)
plot_histogram(statistics)
```

```
{'0': 57, '1': 43}
```

[11]:



7.4.2 Performing operations with “Operator” and “Statevector” :

- Unitary operations can be defined and performed on state vectors in Qiskit using the Operator class, as in the example that follows.

[12]:
```
from qiskit.quantum_info import Operator

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.draw("latex")
```

[12]:

$$|0\rangle$$

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$
$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} T = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix}$$

[13]:
```
v = v.evolve(H)

v.draw("latex")
```

[13]:

$$\frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}}{2}|1\rangle$$

Here hadamard gate ***H*** is applied on v. We know that

$$v = |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$
$$H = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

Now let us try to perform the operation of Hadamard on v:

$$\begin{aligned} H \cdot v &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{\sqrt{2}} + 0 \\ \frac{1}{\sqrt{2}} + 0 \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \\ \implies H \cdot v &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \implies H \cdot v &= \frac{\sqrt{2}}{2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \frac{\sqrt{2}}{2} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \implies H \cdot v &= \frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}}{2}|1\rangle \end{aligned}$$

[14]:
```
v = v.evolve(T)

v.draw("latex")
```

[14]:

$$\frac{\sqrt{2}}{2}|0\rangle + (\frac{1}{2} + \frac{i}{2})|1\rangle$$

now applying T gate to the above response.

$$
\begin{aligned}
T \cdot (H \cdot v) &= \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \\
T \cdot (H \cdot v) &= \begin{pmatrix} \frac{1}{\sqrt{2}} + 0 \\ 0 + \frac{1+i}{2} \end{pmatrix} \\
&= \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1+i}{2} \end{pmatrix} = \frac{1}{\sqrt{2}}|0\rangle + \frac{1+i}{2}|1\rangle \\
\implies T \cdot (H \cdot v) &= \frac{\sqrt{2}}{2}|0\rangle + \frac{1+i}{2}|1\rangle
\end{aligned}
$$

[15]:
```
v = v.evolve(H)

v.draw("latex")
```

[15]:

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

Let's apply the Hadamard gate H to the updated state vector $T.H.v$:

The Hadamard gate matrix is:

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

And the current updated state vector is:

$$T \cdot H \cdot v = \frac{\sqrt{2}}{2}|0\rangle + \frac{1+i}{2}|1\rangle$$

Now, let's perform the matrix-vector multiplication:

$$H \cdot (T \cdot H \cdot v) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \left(\frac{\sqrt{2}}{2}|0\rangle + \frac{1+i}{2}|1\rangle \right)$$

Simplifying this, we get:

$$H \cdot (T \cdot H \cdot v) = \begin{bmatrix} \frac{1}{2} + \frac{1+i}{2\sqrt{2}} \\ \frac{1}{2} - \frac{1+i}{2\sqrt{2}} \end{bmatrix}$$

Combine the terms:

$$H \cdot (T \cdot H \cdot v) = \frac{1}{2} \begin{bmatrix} 1 + \frac{1+i}{\sqrt{2}} \\ 1 - \frac{1+i}{\sqrt{2}} \end{bmatrix}$$

So, after applying the Hadamard gate to the updated state vector, we should obtain:

$$H \cdot (T \cdot H \cdot v) = \frac{1}{2}\left[\left(1 + \frac{1+i}{\sqrt{2}}\right)|0\rangle + \left(1 - \frac{1+i}{\sqrt{2}}\right)|1\rangle\right]$$

and so on..

[16]:
```
v = v.evolve(T)

v.draw("latex")
```

[16]:

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

[17]:
```
v = v.evolve(Z)

v.draw("latex")
```

[17]:

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

7.4.3 Looking ahead toward quantum circuits :

we have Qiskit's **QuantumCircuit** class for creating Quantum circuits.

[18]:
```
from qiskit import QuantumCircuit

circuit = QuantumCircuit(1)

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

circuit.draw()
```

[18]:

```
q:  H  T  H  T  Z
```

This Quantum circuit is the circuit that shows unitary operations performed on State vector **v** in the perivious example.

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.

[19]:
```
ket0 = Statevector([1, 0])
v = ket0.evolve(circuit)
v.draw("latex")
```

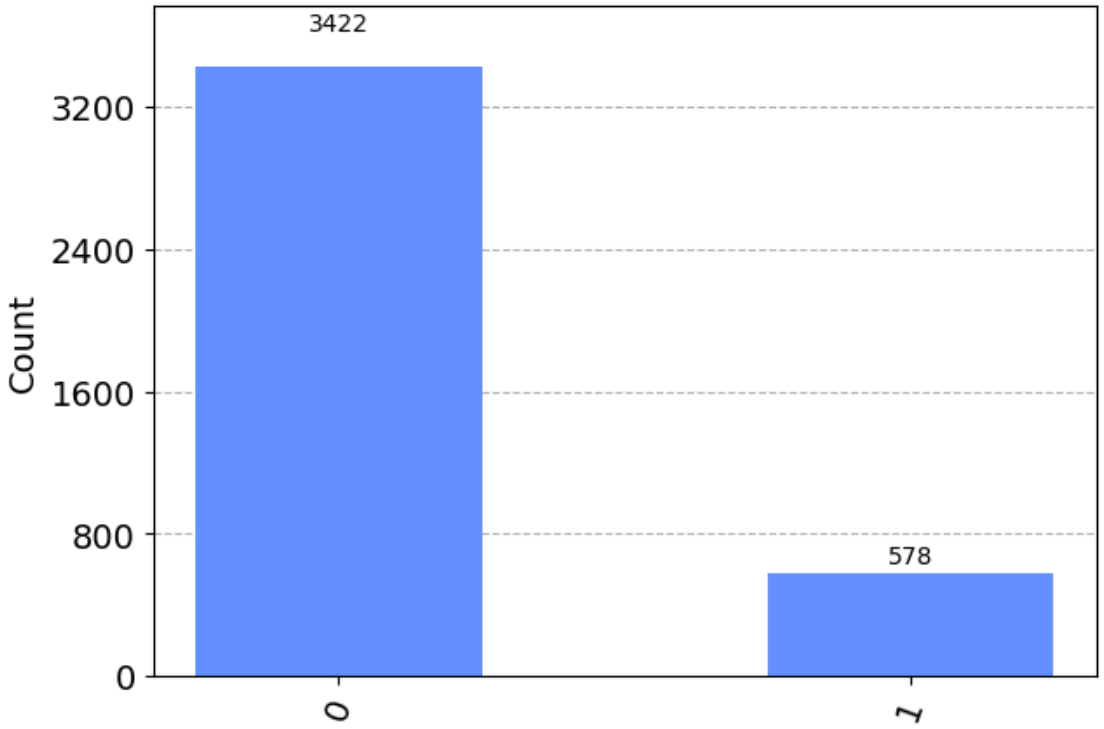
[19]:

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

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.

[20]:
```
statistics = v.sample_counts(4000)
plot_histogram(statistics)
```

[20]:



8 Bloch Sphere Representation :

The Bloch Sphere is a graphical representation used in quantum mechanics to visualize the state of a two-level quantum system, such as a qubit. Here's a simple explanation of the Bloch Sphere:

1. **Qubits and Quantum States:**
   - In classical computing, bits can exist in one of two states: 0 or 1. In quantum computing, we have qubits, which can exist in multiple states simultaneously due to the principles of superposition.
2. **Bloch Sphere Basics:**
   - The Bloch Sphere is a spherical representation where each point on the surface of the sphere corresponds to a possible quantum state of a qubit.

3. **Spherical Coordinates:**
    - The sphere is defined using spherical coordinates: θ (theta) and ϕ (phi).
    - θ represents the **polar angle**, indicating how much the state is aligned with the 'up' axis.
    - ϕ represents the **azimuthal angle**, indicating the rotation around the 'up' axis.
4. **Polar and Azimuthal Angles:**
    - For any qubit state, we can find a unique set of θ and ϕ values that determine its **position** on the Bloch Sphere.
5. **Equator and Poles:**
    - The equator of the sphere represents all possible superpositions of the 0 and 1 states.
    - The poles represent the pure states 0 and 1.
6. **Superposition:**
    - States on the equator are superpositions of 0 and 1, meaning the qubit has some probability of being in both states simultaneously.
7. **Bloch Vector:**
    - The state of a qubit is often represented by a vector called the **Bloch vector**.
    - The length of the vector represents the **probability amplitude** of the qubit being in the 0 or 1 state.
    - The direction of the vector corresponds to the angles θ and ϕ on the Bloch Sphere.
8. **Measurement:**
    - When we measure a qubit, it **collapses** to one of the poles (0 or 1) based on the probabilities determined by the Bloch vector.
9. **Visualization Tool:**

- The Bloch Sphere is a useful tool for visualizing quantum gates and operations, helping to understand how they manipulate the quantum state.

In summary, the Bloch Sphere provides an intuitive way to grasp the complex nature of quantum states, superposition, and measurement in a two-level quantum system like a qubit. It's a handy visualization tool for understanding and working with quantum computing concepts.

A qubit, the fundamental unit of quantum information, can be in a superposition of two basis states: $|0\rangle$ and $|1\rangle$. The state of a qubit is typically represented as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

Where: - $|0\rangle$ and $|1\rangle$ are complex numbers known as probability amplitudes, satisfying the normalization condition

$$(|\alpha|^2 + |\beta|^2 = 1)$$

Bloch Sphere Coordinates:

The qubit state vector $|\psi\rangle$ is associated with three spherical coordinates: θ, ϕ, and the radius $\boldsymbol{r = 1}$.

1. **Radius (r = 1):** The Bloch sphere has a fixed radius of 1, representing the normalization constraint of the qubit state vector.
2. **Angles (and):**
   - θ represents the angle between the state vector and the z-axis (polar angle).
   - ϕ represents the angle in the **xy**-plane measured counterclockwise from the positive x-axis (azimuthal angle).

Operations or gates applied to qubits correspond to transformations on the Bloch sphere. For instance, a Hadamard gate rotates the state vector around specific axes on the sphere.

8.1 General equation of Bloch Sphere :

Pauli matrices form the basis for many quantum gates in quantum computing. Operations such as X-gate (bit-flip), Y-gate (bit and phase-flip), and Z-gate (phase-flip) are represented by these matrices.

The four Pauli matrices are as discussed before:

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

The Bloch Sphere, which is a visual representation of qubit states, relies on Pauli matrices to describe the rotation and manipulation of quantum states.

Each one of the the pauli matrices σ_x,σ_y,σ_z has an eigen values of **+1** corresponding to state $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and **-1** corresponding to state $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Let us try to find out pauli matrix for any arbitrary direction, which is denoted as σ_n. This arbitrary direction has the following vector components:

$$\begin{aligned} \hat{x} &= r \sin\theta \cdot \cos\phi \\ \hat{y} &= r \sin\theta \cdot \sin\phi \\ \hat{z} &= r \cos\theta \end{aligned}$$

If we consider **r=1** that makes σ_n (***Pauli matrix along arbitrary direction n***) as $\sigma_z \cos\theta$. This makes the diagonal elements of σ_n as $\cos\theta$ and $-\cos\theta$.

$$\implies \sigma_z \cos\theta = \begin{bmatrix} \cos\theta & 0 \\ 0 & -\cos\theta \end{bmatrix} \tag{a}$$

Similiarly,

$$\implies \sigma_x(\sin\theta\cos\phi) = \begin{bmatrix} 0 & \sin\theta\cos\phi \\ \sin\theta\cos\phi & 0 \end{bmatrix} \tag{b}$$

$$\implies \sigma_y(\sin\theta\sin\phi) = \begin{bmatrix} 0 & -i\sin\theta\sin\phi \\ i\sin\theta\sin\phi & 0 \end{bmatrix} \tag{c}$$

So therefore, from (1),(2),(3)

$$\implies \sigma_n = \sigma_x \cdot \hat{x} + \sigma_y \cdot \hat{y} + \sigma_z \cdot \hat{z}$$
$$\implies \sigma_n = \begin{bmatrix} 0 & \sin\theta\cos\phi \\ \sin\theta\cos\phi & 0 \end{bmatrix} + \begin{bmatrix} 0 & -i\sin\theta\sin\phi \\ i\sin\theta\sin\phi & 0 \end{bmatrix} + \begin{bmatrix} \cos\theta & 0 \\ 0 & -\cos\theta \end{bmatrix}$$

So the resultant pauli matrix for any arbitrary direction n is given by

$$\boxed{\sigma_n = \begin{bmatrix} \cos\theta & e^{-i\phi}\sin\theta \\ e^{i\phi}\sin\theta & -\cos\theta \end{bmatrix}}$$

The eigen values of σ_n is can be found by:

$$\det[\sigma_n - \lambda \cdot I] = 0$$
$$\implies \det\left[\begin{bmatrix} \cos\theta & e^{-i\phi}\sin\theta \\ e^{i\phi}\sin\theta & -\cos\theta \end{bmatrix} \lambda \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right] = 0$$
$$\implies \det \begin{bmatrix} \cos\theta - \lambda & e^{-i\phi}\sin\theta \\ e^{i\phi}\sin\theta & -\cos\theta - \lambda \end{bmatrix} = 0$$

$$\implies (\cos\theta - \lambda)\cdot(-\cos\theta - \lambda) \; - \; (e^{i\phi}\sin\theta \cdot e^{-i\phi}\sin\theta) \; = 0$$

on further solving we get $\lambda = \pm 1$.

We have seen that eigen values of all Pauli matrices are ± 1. So even if we take any n^{th} direction it also gives the same eigen value ± 1.

Now we can use the same matrix to find out what are eigen states associated with these eigen values ± 1.

Let us calculate the eigen state for the corresponding eigen value **+1** as it is the standard convention for the pictorial representation.

$$eigen\ state\ |\theta,\phi\rangle = \begin{bmatrix} \cos\frac{\theta}{2} \\ e^{i\phi}\sin\frac{\theta}{2} \end{bmatrix}$$

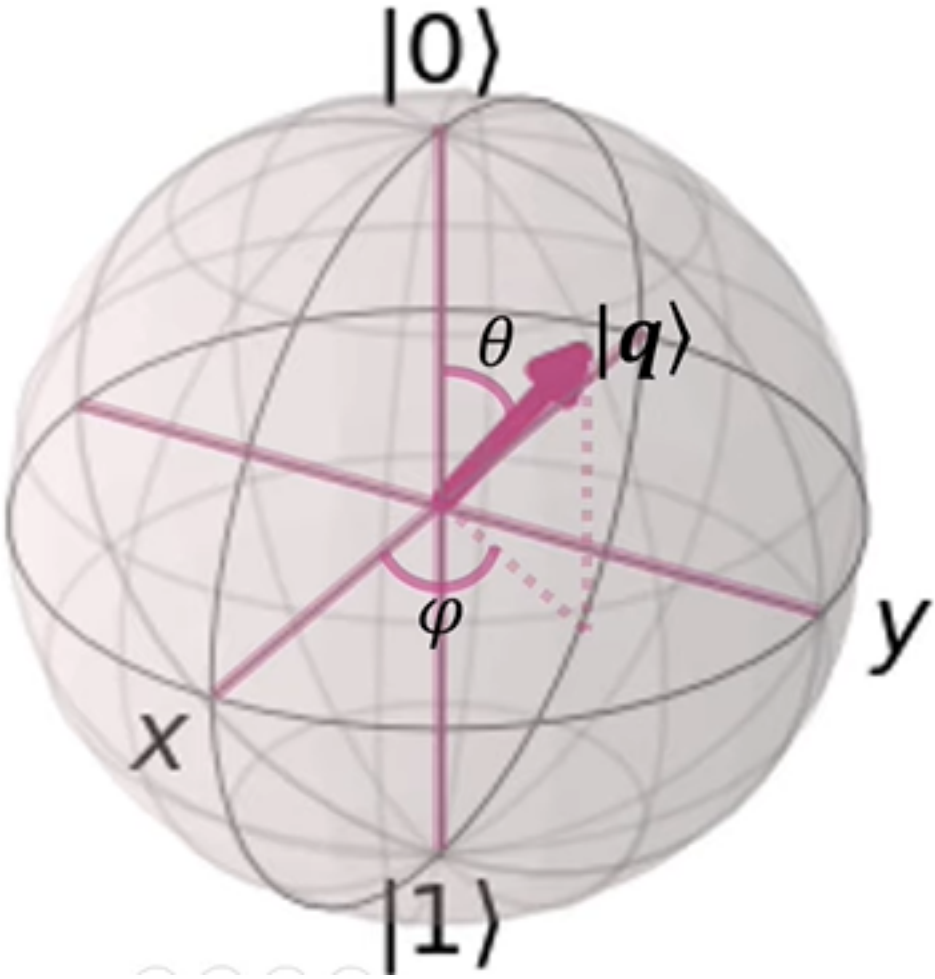
The corresponding pictorial representation of this state is called Bloch sphere which is given below. This Bloch sphere is a sphere of unit radius(so we made **r=1**).

So this can be represented as follows:

$$|\theta,\phi\rangle = \cos\frac{\theta}{2}\begin{bmatrix} 1 \\ 0 \end{bmatrix} + e^{i\phi}\cdot\sin\frac{\theta}{2}\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Similiarly, its vector representation is given as follows:

$$|\theta,\phi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\cdot\sin\frac{\theta}{2}|1\rangle$$



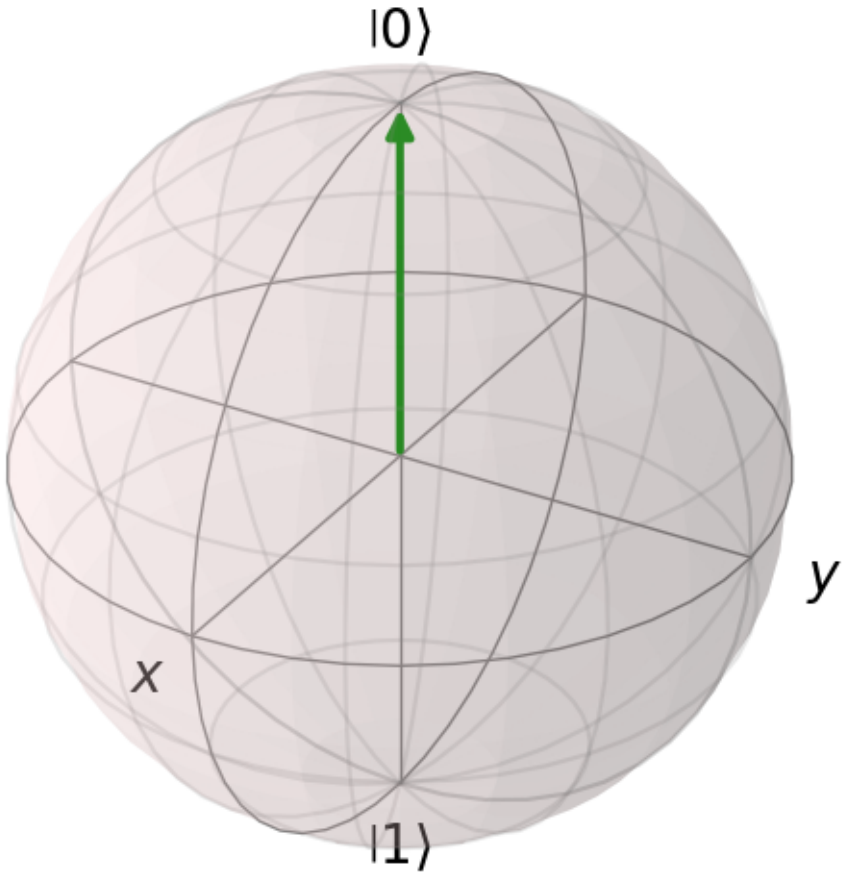
Let us consider a point of $\theta = 0$ and $\phi = 0$ where **r=1**.

8.2 Representation of State-0:

[21]:
```
import numpy as np
import matplotlib.pyplot as plt
from qutip import Bloch

def bloch_sphere(theta, phi):
    # Angles to radians conversion
    theta_rad = theta * np.pi
    phi_rad = phi * 2 * np.pi

    # Calculate Bloch vector components
    x = np.sin(theta_rad) * np.cos(phi_rad)
    y = np.sin(theta_rad) * np.sin(phi_rad)
    z = np.cos(theta_rad)

    print(f"Theta: {theta}, Phi: {phi}")
```

```
    print(f"X: {x}, Y: {y}, Z: {z}")

    # Create a Bloch sphere
    b = Bloch()

    # Add the Bloch vector with arrow indication
    b.add_vectors([x, y, z])

    # Show the Bloch sphere
    b.show()

bloch_sphere(0, 0)
```

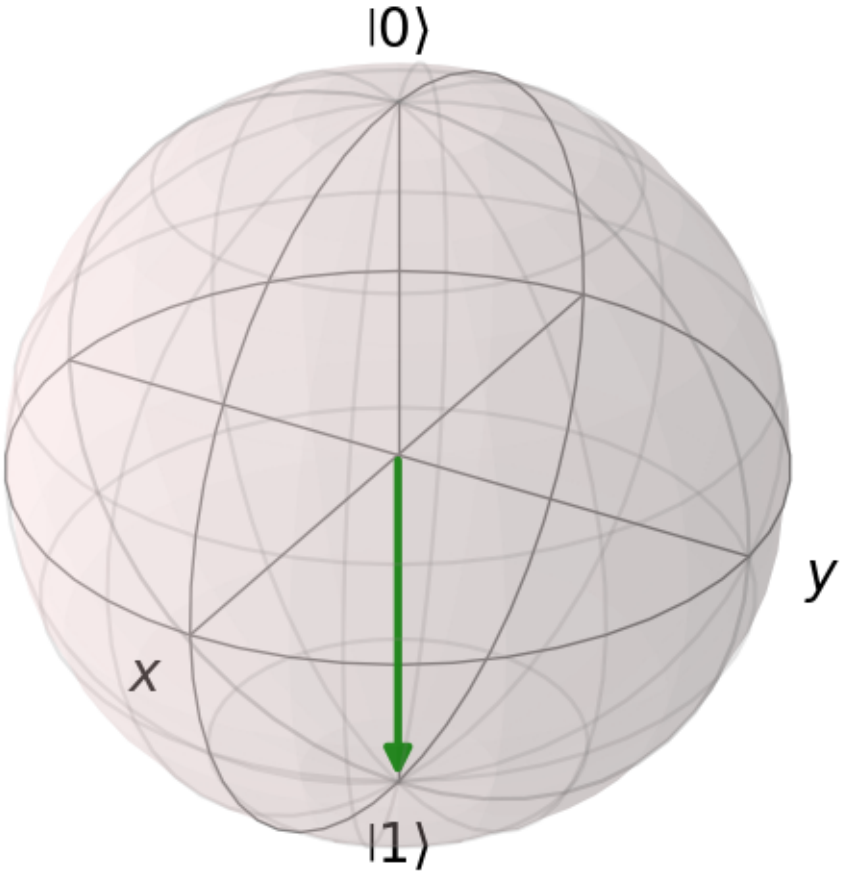
```
Theta: 0, Phi: 0
X: 0.0, Y: 0.0, Z: 1.0
```



8.3 Representation of state-1 :

[22]:
```
import numpy as np
import matplotlib.pyplot as plt
from qutip import Bloch

def bloch_sphere(theta, phi):
    # Angles to radians conversion
    theta_rad = theta * np.pi
    phi_rad = phi * 2 * np.pi

    # Calculate Bloch vector components
    x = np.sin(theta_rad) * np.cos(phi_rad)
    y = np.sin(theta_rad) * np.sin(phi_rad)
    z = np.cos(theta_rad)

    print(f"Theta: {theta}, Phi: {phi}")
    print(f"X: {x}, Y: {y}, Z: {z}")

    # Create a Bloch sphere
    b = Bloch()

    # Add the Bloch vector with arrow indication
    b.add_vectors([x, y, z])

    # Show the Bloch sphere
    b.show()

bloch_sphere(1, 0)
```

```
Theta: 1, Phi: 0
X: 1.2246467991473532e-16, Y: 0.0, Z: -1.0
```



8.4 Representation of Special states :

We know that the vector representation of Bloch sphere is as follows:

$$|\theta, \phi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi} \cdot \sin\frac{\theta}{2}|1\rangle$$

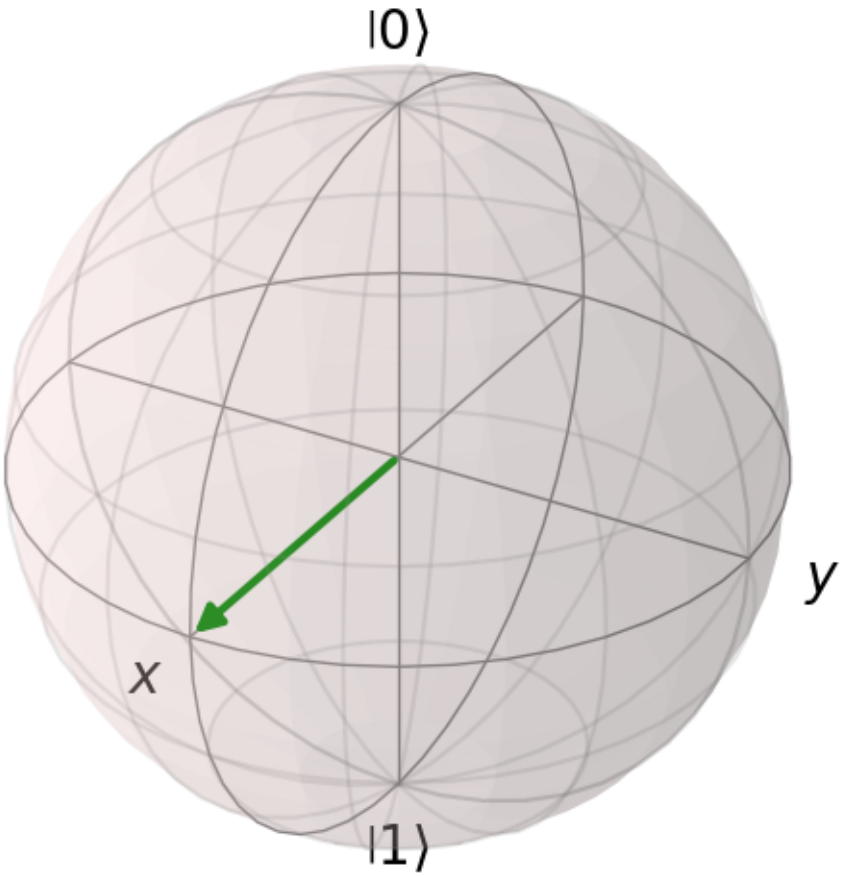
1. When $\theta = \frac{\pi}{2}$ and $\phi = 0$: (Plot along positive x-axis)

$$\implies \frac{1}{\sqrt{2}}[|0\rangle + |1\rangle] = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

[23]:
```
def bloch_sphere(theta, phi):
    # Angles to radians conversion
    theta_rad=np.deg2rad(theta)
    phi_rad = np.deg2rad(phi)
```

```
    # Calculate Bloch vector components
    x = np.sin(theta_rad) * np.cos(phi_rad)
    y = np.sin(theta_rad) * np.sin(phi_rad)
    z = np.cos(theta_rad)

    print(f"Theta: {theta}, Phi: {phi}")
    print(f"X: {x}, Y: {y}, Z: {z}")

    # Create a Bloch sphere
    b = Bloch()

    # Add the Bloch vector with arrow indication
    b.add_vectors([x, y, z])

    # Show the Bloch sphere
    b.show()


bloch_sphere(90,0)
#input should be passed as the angles.
#Using np.pi takes the value of pi (in radians) which gives incorrect result.
```

```
Theta: 90, Phi: 0
X: 1.0, Y: 0.0, Z: 6.123233995736766e-17
```

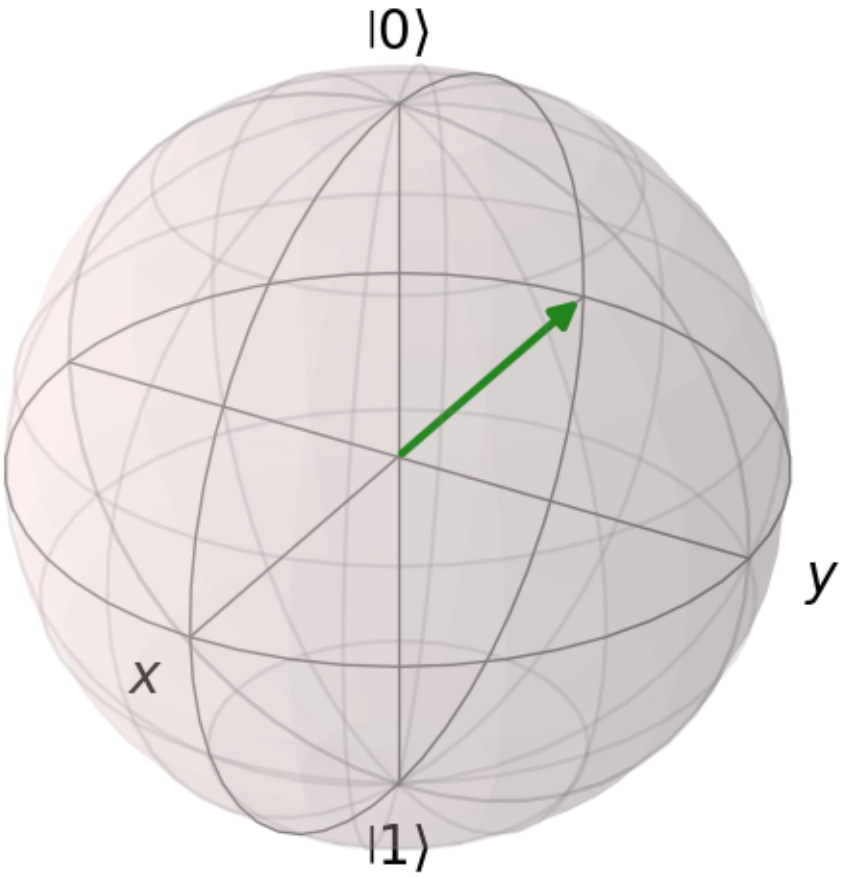


2. When $\theta = \frac{\pi}{2}$ and $\phi = \pi$: (Plot along negative x-axis)

$$\implies \frac{1}{\sqrt{2}}[|0\rangle - |1\rangle] = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{bmatrix}$$

[24]:
```
bloch_sphere(90,180)
```

```
Theta: 90, Phi: 180
X: -1.0, Y: 1.2246467991473532e-16, Z: 6.123233995736766e-17
```

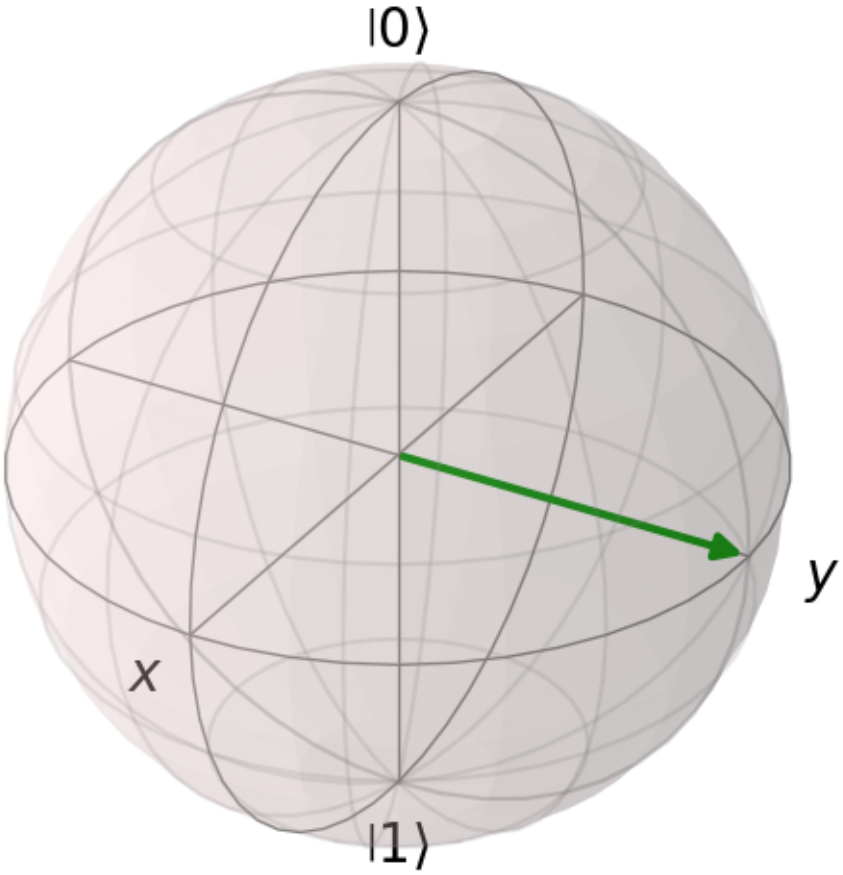


3. When $\theta = \frac{\pi}{2}$ and $\phi = \pi$: (Plot along positive y-axis)

$$\implies |+i\rangle = \frac{1}{\sqrt{2}}[|0\rangle + i|1\rangle] = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} \end{bmatrix}$$

[25]:
```
bloch_sphere(90,90)
```

```
Theta: 90, Phi: 90
X: 6.123233995736766e-17, Y: 1.0, Z: 6.123233995736766e-17
```



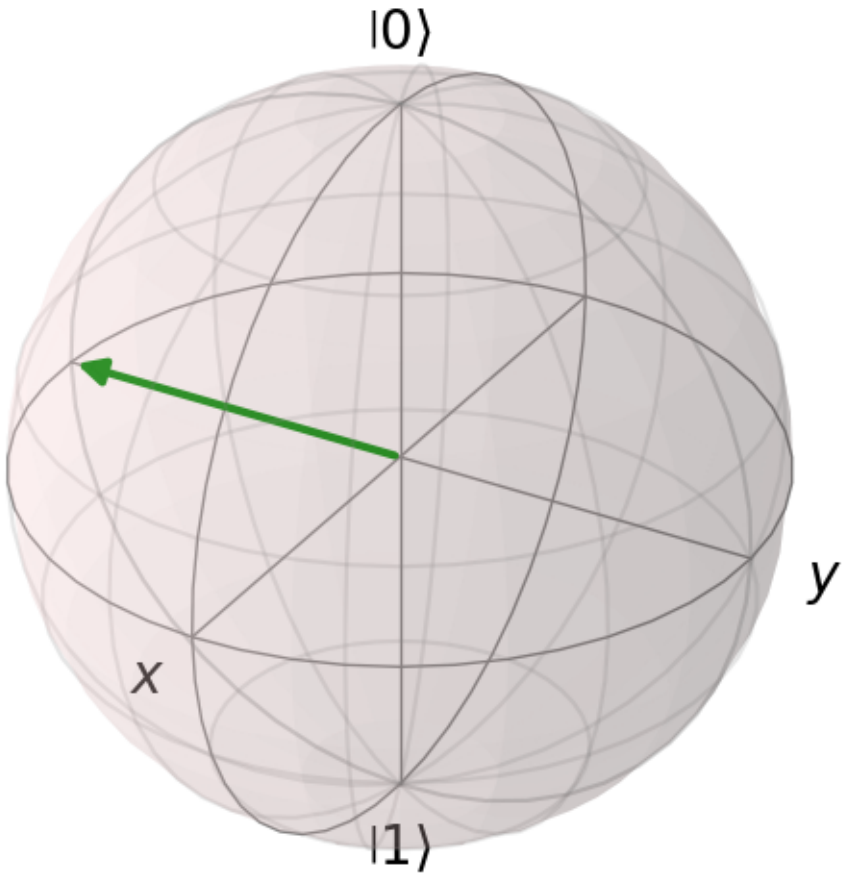
4. When $\theta = \frac{-\pi}{2}$ and $\phi = \frac{\pi}{2}$: (Plot along negative y-axis)

$$\implies |-i\rangle = \frac{1}{\sqrt{2}}[|0\rangle - i|1\rangle] = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{-i}{\sqrt{2}} \end{bmatrix}$$

We can also use $\theta = \frac{\pi}{2}$ and $\phi = \frac{3\pi}{2}$ combination for the same response.

[26]:
```
bloch_sphere(-90,90)
```

```
Theta: -90, Phi: 90
X: -6.123233995736766e-17, Y: -1.0, Z: 6.123233995736766e-17
```



On the Bloch Sphere, angles are twice as big as in Hilbert Space.

- For example $|0\rangle$and $|1\rangle$ are orthogonal in Hilbert Space i.e., angle between them is 90-degrees.
- But where as in Bloch sphere, the angle between states $|0\rangle$ and $|1\rangle$ is 180-degrees.
- If we consider the values of they are the values corresponding to Bloch Sphere where as the actual values (at Hilbert space) are $\frac{}{\mathbf{2}}$.

9 Information in Multiple Systems :

The focus of this lesson is on the basics of quantum information when there are **multiple** systems being considered. Such situations arise naturally in the context of information processing, both classical and quantum. Large information-carrying systems are often most easily constructed using collections of smaller systems, such as bits or qubits.

- Here we always choose to view multiple systems together as if they form a single, compound system - to which all the concepts of states and measurements discussed for Single Systems

can be applied.

- This idea very directly leads to a description of how quantum states, measurements, and operations work for multiple systems.
- There is a requirement to have better understanding to the compound system rather than considering the whole system as one system.For instance, we may have multiple quantum systems that are collectively in a particular quantum state, and then choose to measure just one (or a proper subset) of the individual systems.
- In general this will affect the states of remaining systems. So, understanding of the all sorts of correlations among multiple systems — and particularly a type of correlation known as **entanglement** — is also important in quantum information and computation.

9.1 Classical Information in Multiple Systems :

9.1.1 Classical states via the Cartesian product :

We will start at a very basic level, with classical states of multiple systems. For simplicity, we will begin by discussing just two systems, and then generalize to more than two systems.

To be precise, let us suppose that X is a system whose classical state set is Σ, and Y is a second system having classical state set Γ.

- As in the previous lesson, because we have referred to these sets as classical state sets, our assumption is that Σ and Γ are both **finite** and **nonempty**.
- It could be that $\Sigma = \Gamma$, but this is not necessarily so — and regardless it is helpful to use different names to refer to these sets in the interest of clarity.

Now imagine that the two systems, X and Y, are placed side-by-side, with X on the left and Y on the right.

- If we so choose, we can view these two systems as if they form a single system, which we can denote by (X,Y) or XY depending on our preference.

A natural question to ask about this compound system (X,Y) is, ***"What are its classical states?"***

- The answer is that the set of classical states of (X,Y) is the **Cartesian product** of Σ and Γ, which is the set defined as

$$\boxed{\Sigma \times \Gamma = \big\{(a,b) \,:\, a \in \Sigma \text{ and } b \in \Gamma\big\}}$$

In simple terms, the Cartesian product is precisely the mathematical notion that captures the idea of viewing an element of one set and an element of a second set together, as if they form a single element of a single set.

In this case, to say that (X,Y) is in the classical state $\in \Sigma \times \Gamma$ means that

- X is in the classical state $a \in \Sigma$ and
- Y is in the classical state $b \in \Gamma$; and
- if the classical state of X is $a \in \Sigma$ and the classical state of Y is $b \in \Gamma$, then the classical state of the joint system (X,Y) is (a,b).

Example :

- If Σ={0,1} and Γ={♣,♢,♡,♠}, then

$$\boxed{\Sigma \times \Gamma \ = \{(0,\clubsuit),(0,\diamondsuit),(0,\heartsuit),(0,\spadesuit),(1,\clubsuit),(1,\diamondsuit),(1,\heartsuit),(1,\spadesuit)\}}$$

For more than two systems, the situation generalizes in a natural way. If we suppose that $\mathsf{X}_1, \ldots, \mathsf{X}_n$ where, X are systems having classical state sets $\Sigma_1, \ldots, \Sigma_n$, respectively, for any positive integer n, the classical state set of the $n-tuple$ $(\mathsf{X}_1, \ldots, \mathsf{X}_n)$, viewed as a **single joint system**, is the Cartesian product

$$\Sigma_1 \times \cdots \times \Sigma_n = \{(a_1, \ldots, a_n) \,:\, a_1 \in \Sigma_1,\ \ldots,\ a_n \in \Sigma_n\}$$

Example :

- if$\Sigma_1 \ = \Sigma_2 \ = \Sigma 3 = \{0,1\}$, then the classical state set of $(\mathsf{X}_1, \mathsf{X}_2, \mathsf{X}_3)$ is

$$\boxed{\Sigma_1 \times \Sigma_2 \times \Sigma 3 = \{(0,0,0),(0,0,1),(0,1,0),(0,1,1),(1,0,0),(1,0,1),(1,1,0),(1,1,1)\}}$$

9.1.2 Representing States as a string :

It is often convenient to write a classical state $(a_1, \ldots, a_n)$ as a string $a_1 \cdots a_n$ for the sake of brevity, particularly in the (very typical) situation that the classical state sets $\Sigma_1, \ldots, \Sigma_n$ are associated with sets of *symbols* or *characters*.

Indeed, the notion of a string, which is a fundamentally important concept in computer science, is formalized in mathematical terms through Cartesian products. The term alphabet is commonly used to refer to sets of symbols used to form strings, but the mathematical definition of an alphabet is precisely the same as the definition of a classical state set: **it is a finite and nonempty set**.

For example, suppose that $\mathsf{X}_1, \ldots, \mathsf{X}_{10}$ are bits, so that the classical state sets of these systems are all the same.

$$\Sigma_1 = \Sigma_2 = \cdots = \Sigma_{10} = \{0,1\}$$

(The set $\{0,1\}$ is commonly referred to as the binary alphabet.) There are then $2^{10} = 1024$ classical states of the joint system $(\mathsf{X}_1, \ldots, \mathsf{X}_{10})$, which are the elements of the set

$$\Sigma_1 \times \Sigma_2 \times \cdots \times \Sigma_{10} = \{0,1\}^{10}$$

Written as strings, these classical states look like this:

0000000000
0000000001
0000000010
0000000011
0000000100
⋮
1111111111

For the classical state **0001010000**, for instance, we see that X_4 and X_6 are in the state 1, while all other systems are in the state 0.

Cartesian Products of classical states are ordered **lexicographically** (i.e., dictionary ordering).

- We assume individual classical state sets are already ordered.
- Significance decreases from left to right.

Example : The cartesian product **{1,2,3}×{0,1}** is ordered like this :

$$(1,0),(1,1),(2,0),(2,1),(3,0),(3,1)$$

When n-tuples are written as strings and ordered in this way, we observe familiar patterns, such as {0,1} × {0,1} being ordered as 00,01,10,11.

9.1.3 Probabilistic States :

We know that a ***probabilistic state*** associates a **probability with each classical state of a system**. Thus, a probabilistic state of multiple systems — viewed collectively as if they form a single system — associates a probability with each element of the Cartesian product of the classical state sets of the individual systems.

For example, suppose that X and Y are both bits, so that their corresponding classical state sets are $\Sigma = \{0,1\}$ and $\Gamma = \{0,1\}$, respectively. Here is a probabilistic state of the pair (X,Y):

$$\begin{aligned}
&\Pr((\mathsf{X},\mathsf{Y}) = (0,0)) = 1/2\\
&\Pr((\mathsf{X},\mathsf{Y}) = (0,1)) = 0\\
&\Pr((\mathsf{X},\mathsf{Y}) = (1,0)) = 0\\
&\Pr((\mathsf{X},\mathsf{Y}) = (1,1)) = 1/2
\end{aligned}$$

This probabilistic state is one in which both X and Y are random bits — each is **0** with probability $1/2$ and **1** with probability $1/2$ — but the classical states of the two bits always agree. This is an example of a correlation between these systems.

Ordering Cartesian product state sets : Probabilistic states of systems are represented by probability vectors, which are column vectors having indices that have been placed in correspondence with the underlying classical state set of the system being considered.

The same situation arises for multiple systems. To represent a probabilistic state of multiple systems as a Cartesian product, one must decide on an ordering of the product's elements. Assuming the individual classical state sets Σ, Γ of systems X, Y are already ordered, there is a simple convention for doing this: alphabetical ordering.

More precisely, the entries in each n-tuple (or, equivalently, the symbols in each string) are viewed as being ordered by significance that decreases from left to right.

For example, according to this convention, the Cartesian product $\{1,2,3\} \times \{0,1\}$ is ordered like this:

$$(1,0),\ (1,1),\ (2,0),\ (2,1),\ (3,0),\ (3,1)$$

When n-tuples are written as strings and ordered in this way, we observe familiar patterns, such as $\{0,1\} \times \{0,1\}$ being ordered as $00, 01, 10, 11$, and the set $\{0,1\}^{10}$ being ordered as was suggested above.

We also see $\{0,1,\ldots,9\} \times \{0,1,\ldots,9\}$ ordered as the numbers **0** through 99. You may recognize this is not a coincidence: today's decimal number system uses the same alphabetical ordering. Here, of course, "alphabetical" has a broader meaning that may include a collection of numeric symbols.

Returning to the example of two bits from above, the probabilistic state is represented by the following probability vector (where the entries are labeled explicitly for the sake of clarity).

$$\begin{pmatrix} \frac{1}{2} \\ 0 \\ 0 \\ \frac{1}{2} \end{pmatrix} \begin{array}{l} \leftarrow \text{probability associated with state 00} \\ \leftarrow \text{probability associated with state 01} \\ \leftarrow \text{probability associated with state 10} \\ \leftarrow \text{probability associated with state 11} \end{array} \tag{1}$$

Here the probability associated with states 01 and 10 is explicitly considered as 0. This is not an assumption but a specific choice made to illustrate the concept of independence.

The assumption of 0 probability for **01** and **10** is tied to the idea that, in an independent system, the occurrence of one state *should not provide any information* about the occurrence of the other state.

In other words, if X and Y are **independent**, knowing the state of X tells us nothing about the state of Y and vice versa.

Consider the probability vector provided above:

$$\begin{pmatrix} \frac{1}{2} \\ 0 \\ 0 \\ \frac{1}{2} \end{pmatrix}$$

This implies that X and Y are independent because the probabilities of **01** and **10** are explicitly set to 0.

If, for example, the probability of **01** was **non-zero**, it would mean that observing $\mathsf{X} = 0$ gives information about the likelihood of $\mathsf{Y} = 1$, indicating a form of correlation between X and Y.

So, the choice of 0 probability for **01** and **10** is a deliberate representation to demonstrate the concept of independence between the two bits.

Example :

Let's consider a simplified scenario with two coins, each having two possible states: Heads (H) or Tails (T). The classical state sets for the two coins are $\{H, T\}$.

The Cartesian product of the state sets is $\{H, T\} \times \{H, T\}$, and when ordered alphabetically, we get the following list of ordered pairs:

$$(H,H), (H,T),$$
$$(T,H), (T,T).$$

Now, let's assume we have a probabilistic state represented by the probability vector:

$$\begin{pmatrix} \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \\ \frac{1}{4} \end{pmatrix}$$

Here, each entry corresponds to the probability associated with the ordered pair in the Cartesian product. The ordering follows the alphabetical convention.

This probability vector represents the probabilistic state of the two coins. Each entry in the vector indicates the probability of obtaining the corresponding outcome when the two coins are flipped. For example, the probability of getting (H, T) is $\frac{1}{4}$.

In this simplified example, the alphabetical ordering is straightforward, making it easy to see how the probability vector aligns with the different outcomes of flipping two coins.

Independence of two systems : Independence:

A special type of probabilistic state of two systems is one in which the systems are **independent**.

- Intuitively speaking, two systems are independent if learning the classical state of either system has no effect on the probabilities associated with the other.
- That is, learning what classical state one of the systems is in provides no information at all about the classical state of the other. (**No correlation**)

Representation using Probability Vectors:

To define this notion precisely, let us suppose once again that X and Y are systems having classical state sets Σ and Γ, respectively. - With respect to a given probabilistic state of these systems, they are said to be independent if it is the case that:

$$Pr((\mathsf{X},\mathsf{Y}) = (a,b)) = Pr(\mathsf{X} = a)Pr(\mathsf{Y} = b) \tag{2}$$

for every choice of $a \in \Sigma$ and $b \in \Gamma$.

- To express this condition in terms of probability vectors, assume that the given probabilistic state of (X,Y) is described by a probability vector, written in the **Dirac notation** as

$$\sum_{(a,b)\in\Sigma\times\Gamma} p_{ab}|ab\rangle$$

- This equation essentially says that the joint probability of the systems being in states (a,b) is the product of their individual probabilities of being in states a and b.
- The condition **(2)** for independence is then equivalent to the existence of two probability vectors

$$|\phi\rangle = \sum_{a\in\Sigma} q_a|a\rangle \quad \text{and} \quad |\psi\rangle = \sum_{b\in\Gamma} r_b|b\rangle, \tag{3}$$

representing the probabilities associated with the classical states of X and Y, respectively, such that

$$p_{ab} = q_a r_b \tag{4}$$

for all $a \in \Sigma$ and $b \in \Gamma$.

Example of Independence:

For example, the probabilistic state of a pair of bits (X, Y) represented by the vector

$$\frac{1}{6}|00\rangle + \frac{1}{12}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{4}|11\rangle$$

is one in which X and Y are independent.

Specifically, the condition required for independence is true for the probability vectors

$$|\phi\rangle = \frac{1}{4}|0\rangle + \frac{3}{4}|1\rangle \quad \text{and} \quad |\psi\rangle = \frac{2}{3}|0\rangle + \frac{1}{3}|1\rangle$$

For example, to match the 00 entry, we need $\frac{1}{6} = \frac{1}{4} \times \frac{2}{3}$, and indeed this is the case. Other entries can be verified in a similar manner.

Correlation as Lack of Independence:

On the other hand, the probabilistic state **(1)**, which we may write as

$$\frac{1}{2}|00\rangle + \frac{1}{2}|11\rangle, \tag{5}$$

does not represent independence between the systems X and Y. A simple way to argue this is as follows.

- The argument is made that for independent systems, certain products of probabilities must be zero, and in this case, it leads to a contradiction. Therefore, the systems are not independent and are defined as correlated.

Suppose that there did exist probability vectors $|\phi\rangle$ and $|\psi\rangle$, as in equation **(3)** above, for which the condition **(4)** is satisfied for every choice of a and b. It would then necessarily be that

$$q_0 r_1 = \Pr((\mathsf{X}, \mathsf{Y}) = (0, 1)) = 0$$

This implies that either $q_0 = 0$ or $r_1 = 0$, because if both were nonzero, the product $q_0 r_1$ would also not be zero. This leads to the conclusion that either $q_0 r_0 = 0$ (in case $q_0 = 0$) or $q_1 r_1 = 0$ (in case $r_1 = 0$).

We see, however, that neither of those equalities can be true because we must have $q_0 r_0 = 1/2$ and $q_1 r_1 = 1/2$. Hence, there do not exist vectors $|\phi\rangle$ and $|\psi\rangle$ satisfying the property required for independence.

Having defined independence between two systems, we can now define correlation precisely as a lack of independence. For example, because the two bits in the probabilistic state represented by the vector **(5)** are **not independent**, they are, by definition, **correlated**.

Example :

Let's consider a simple example involving two six-sided dice. The classical states of each die are the numbers 1 through 6.

Suppose we have two dice, and we're interested in the joint probabilistic state when both dice are rolled. If the dice are independent, the joint probability of getting a specific outcome, say (3, 5),

should be the product of the individual probabilities of getting a 3 on the first die and a 5 on the second die.

Let's define the probability vectors for each die:

- For the first die, let's say $|\phi\rangle = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$.
- For the second die, let $|\psi\rangle = \frac{1}{6}|1\rangle + \frac{1}{6}|2\rangle + \frac{1}{6}|3\rangle + \frac{1}{6}|4\rangle + \frac{1}{6}|5\rangle + \frac{1}{6}|6\rangle$.

Now, if the dice are independent, the joint probability vector for both dice should be the **tensor product** of $|\phi\rangle$ and $|\psi\rangle$:

$$|\Phi\rangle = |\phi\rangle \otimes |\psi\rangle.$$

In this case, the joint probability of getting a (3, 5) when both dice are rolled is:

$$\langle 3|\phi\rangle \cdot \langle 5|\psi\rangle = \frac{1}{6} \times \frac{1}{6} = \frac{1}{36}.$$

This is the product of the individual probabilities of getting a 3 on the first die and a 5 on the second die.

If the joint probability for all possible outcomes follows this pattern, then the dice are considered independent. If not, they are correlated.

let's consider a case where the two dice are correlated. In this scenario, the outcome of one die provides information about the other die.

Suppose the joint probabilistic state is given by:

$$|\Phi_{\text{correlated}}\rangle = \frac{1}{12}|1\rangle \otimes |1\rangle + \frac{1}{12}|2\rangle \otimes |2\rangle + \frac{1}{12}|3\rangle \otimes |3\rangle + \frac{1}{12}|4\rangle \otimes |4\rangle + \frac{1}{12}|5\rangle \otimes |5\rangle + \frac{7}{12}|6\rangle \otimes |6\rangle.$$

Here, notice that the probabilities are not simply the product of the individual probabilities. For example, the probability of getting a (6, 6) is higher than if the dice were independent.

Now, let's calculate the joint probability of getting a (3, 5) in this correlated scenario:

$$\langle 3|\phi\rangle \cdot \langle 5|\psi\rangle = \frac{1}{12} \times \frac{1}{6} = \frac{1}{72}.$$

Unlike in the independent case, the joint probability for (3, 5) is different from the product of the individual probabilities. This indicates that the outcome of one die affects the probabilities of the other, and the dice are considered correlated.

In summary, when the joint probabilities cannot be expressed as the product of individual probabilities, the systems are correlated. In the case of the dice, this means that the outcome of one die provides information about the likely outcome of the other die.

Tensor Product of two Vectors : The condition of independence just described can be expressed more succinctly through the notion of a tensor product. Although this is a very general notion that can be defined quite abstractly and applied to a variety of mathematical structures, in the case at hand it can be defined in simple, concrete terms. Given two vectors

$$|\phi\rangle = \sum_{a\in\Sigma} \alpha_a |a\rangle \quad \text{and} \quad |\psi\rangle = \sum_{b\in\Gamma} \beta_b |b\rangle,$$

the tensor product $|\phi\rangle \otimes |\psi\rangle$ is a new vector over the joint state set $\Sigma \times \Gamma$, defined as

$$\boxed{|\phi\rangle \otimes |\psi\rangle = \sum_{(a,b)\in\Sigma\times\Gamma} \alpha_a \beta_b |ab\rangle}$$

Equivalently, the vector $|\pi\rangle = |\phi\rangle \otimes |\psi\rangle$ is defined by the equation

$$\langle ab|\pi\rangle = \langle a|\phi\rangle\langle b|\psi\rangle$$

being true for every $a \in \Sigma$ and $b \in \Gamma$.

We can now recast the condition for independence as requiring the probability vector $|\pi\rangle$ of the joint system (X, Y) to be representable as a tensor product

$$|\pi\rangle = |\phi\rangle \otimes |\psi\rangle$$

of probability vectors $|\phi\rangle$and $|\psi\rangle$ on each of the subsystems X and Y. - In this situation it is said that $|\pi\rangle$ is a ***product state or product vector***.

We often omit the symbol $\otimes$ when taking the tensor product of kets, such as writing $|\phi\rangle|\psi\rangle$ rather than $|\phi\rangle \otimes |\psi\rangle$. This convention captures the idea that the tensor product is, in this context, the most natural or default way to take the product of two vectors. Although it is less common, the notation $|\phi \otimes \psi\rangle$ is also sometimes used.

Alternative notations :

$$\begin{aligned} |\phi\rangle|\psi\rangle &= |\phi\rangle \otimes |\psi\rangle \\ |\phi \otimes \psi\rangle &= |\phi\rangle \otimes |\psi\rangle \end{aligned}$$

When we use the alphabetical convention for ordering elements of Cartesian products, we obtain the following specification for the tensor product of two column vectors.

$$\begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{pmatrix} \otimes \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_k \end{pmatrix} = \begin{pmatrix} \alpha_1\beta_1 \\ \vdots \\ \alpha_1\beta_k \\ \alpha_2\beta_1 \\ \vdots \\ \alpha_2\beta_k \\ \vdots \\ \alpha_m\beta_1 \\ \vdots \\ \alpha_m\beta_k \end{pmatrix}$$

Example of Tensor product of two vectors :

$$\begin{pmatrix}\alpha_1\\ \alpha_2\\ \alpha_3\end{pmatrix}\otimes\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\\ \beta_4\end{pmatrix}=\begin{pmatrix}\alpha_1\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\\ \beta_4\end{pmatrix}\\ \alpha_2\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\\ \beta_4\end{pmatrix}\\ \alpha_3\begin{pmatrix}\beta_1\\ \beta_2\\ \beta_3\\ \beta_4\end{pmatrix}\end{pmatrix}=\begin{pmatrix}\begin{pmatrix}\alpha_1\beta_1\\ \alpha_1\beta_2\\ \alpha_1\beta_3\\ \alpha_1\beta_4\end{pmatrix}\\ \begin{pmatrix}\alpha_2\beta_1\\ \alpha_2\beta_2\\ \alpha_2\beta_3\\ \alpha_2\beta_4\end{pmatrix}\\ \begin{pmatrix}\alpha_3\beta_1\\ \alpha_3\beta_2\\ \alpha_3\beta_3\\ \alpha_3\beta_4\end{pmatrix}\end{pmatrix}=\begin{pmatrix}\alpha_1\beta_1\\ \alpha_1\beta_2\\ \alpha_1\beta_3\\ \alpha_1\beta_4\\ \alpha_2\beta_1\\ \alpha_2\beta_2\\ \alpha_2\beta_3\\ \alpha_2\beta_4\\ \alpha_3\beta_1\\ \alpha_3\beta_2\\ \alpha_3\beta_3\\ \alpha_3\beta_4\end{pmatrix}$$

As an important aside, we observe the following expression for tensor products of standard basis vectors:

$$|a\rangle\otimes|b\rangle=|ab\rangle$$

Alternatively, writing (a,b) as an ordered pair rather than a string, we could write

$$|a\rangle\otimes|b\rangle=|(a,b)\rangle,$$

but it is more common to write

$$|a\rangle\otimes|b\rangle=|a,b\rangle$$

following a practice in mathematics of removing parentheses that do not add clarity or remove ambiguity.

The tensor product of two vectors has the important property that it is **bilinear**, which means that it is linear in each of the two arguments separately, assuming that the other argument is fixed. This property can be expressed through these equations:

1. Linearity in the first argument:

$$\boxed{\begin{aligned}(|\phi_1\rangle+|\phi_2\rangle)\otimes|\psi\rangle&=|\phi_1\rangle\otimes|\psi\rangle+|\phi_2\rangle\otimes|\psi\rangle\\ ](\alpha|\phi\rangle)\otimes|\psi\rangle&=\alpha(|\phi\rangle\otimes|\psi\rangle)\end{aligned}}$$

2. Linearity in the second argument:

$$\boxed{\begin{aligned}|\phi\rangle\otimes(|\psi_1\rangle+|\psi_2\rangle)&=|\phi\rangle\otimes|\psi_1\rangle+|\phi\rangle\otimes|\psi_2\rangle\\ ]|\phi\rangle\otimes(\alpha|\psi\rangle)&=\alpha(|\phi\rangle\otimes|\psi\rangle)\end{aligned}}$$

Considering the second equation in each of these pairs of equations, we see that scalars "float freely" within tensor products:

$$\boxed{(\alpha|\phi\rangle)\otimes|\psi\rangle=|\phi\rangle\otimes(\alpha|\psi\rangle)=\alpha(|\phi\rangle\otimes|\psi\rangle)}$$

There is therefore no ambiguity in simply writing $\alpha|\phi\rangle\otimes|\psi\rangle$, or alternatively $\alpha|\phi\rangle|\psi\rangle$ or $\alpha|\phi\otimes\psi\rangle$, to refer to this vector.

Independence and tensor products for three or more systems : The notions of independence and tensor products generalize straightforwardly to three or more systems.

If $\mathsf{X}_1, \ldots, \mathsf{X}_n$ are systems having classical state sets $\Sigma_1, \ldots, \Sigma_n$, respectively, then a probabilistic state of the combined system $(\mathsf{X}_1, \ldots, \mathsf{X}_n)$ is a ***product state*** if the associated probability vector takes the form

$$|\psi\rangle = |\phi_1\rangle \otimes \cdots \otimes |\phi_n\rangle$$

for probability vectors $|\phi_1\rangle, \ldots, |\phi_n\rangle$ describing probabilistic states of $\mathsf{X}_1, \ldots, \mathsf{X}_n$.

Here, the definition of the tensor product generalizes in a natural way: the vector

$$|\psi\rangle = |\phi_1\rangle \otimes \cdots \otimes |\phi_n\rangle$$

is defined by the equation

$$\langle a_1 \cdots a_n | \psi \rangle = \langle a_1 | \phi_1 \rangle \cdots \langle a_n | \phi_n \rangle$$

being true for every $a_1 \in \Sigma_1, \ldots a_n \in \Sigma_n$.

A different, but equivalent, way to define the tensor product of three or more vectors is recursively in terms of tensor products of two vectors:

$$|\phi_1\rangle \otimes \cdots \otimes |\phi_n\rangle = \left(|\phi_1\rangle \otimes \cdots \otimes |\phi_{n-1}\rangle\right) \otimes |\phi_n\rangle,$$

assuming $n \geq 3$.

Similar to the tensor product of just two vectors, the tensor product of three or more vectors is linear in each of the arguments individually, assuming that all other arguments are fixed. In this case, we say that the tensor product of three or more vectors is multilinear.

As we did in the case of two systems, we could say that the systems $\mathsf{X}_1, \ldots, \mathsf{X}_n$ are independent when they are in a product state, but the term mutually independent is more precise.

- There happen to be other notions of independence for three or more systems, such as ***pairwise independence***, that we will not be concerned with at this time.

Generalizing the observation earlier concerning tensor products of standard basis vectors, for any positive integer n and any classical states $a_1, \ldots, a_n$ we have

$$|a_1\rangle \otimes \cdots \otimes |a_n\rangle = |a_1 \cdots a_n\rangle = |a_1, \ldots, a_n\rangle.$$

Example :

Let's use a simple example with three classical systems represented by three bits, $\mathsf{X}_1, \mathsf{X}_2$, and X_3, each having classical state sets Σ_1, Σ_2, and Σ_3 respectively.

In the context of a single bit, the standard basis vectors are $|0\rangle$ and $|1\rangle$. Extending this to three bits, the standard basis vectors are:

$$|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle$$

Now, let's consider the tensor product of three standard basis vectors:

$$|0\rangle \otimes |0\rangle \otimes |0\rangle = |000\rangle$$

Similarly:

$$|1\rangle \otimes |0\rangle \otimes |1\rangle = |101\rangle$$

This is another example of the tensor product. In general, for classical states $a_1, a_2, a_3 $, the tensor product:

$$|a_1\rangle \otimes |a_2\rangle \otimes |a_3\rangle = |a_1 a_2 a_3\rangle$$

This notation is a concise way of representing the joint state of the three systems. For instance, $|101\rangle$ represents that X_1 is in state 1, X_2 is in state 0, and X_3 is in state 1.

Now, when we talk about a product state involving probability vectors $|\phi_1\rangle, |\phi_2\rangle$, and $|\phi_3\rangle$, the corresponding joint probabilistic state is a tensor product:

$$|\psi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle \otimes |\phi_3\rangle$$

This would be a product state where the probability associated with each joint state is the product of the probabilities associated with the individual states, similar to what was described earlier for two systems.

9.1.4 Measurements of Probabilistic States in Multiple Systems :

Now let us move on to measurements of probabilistic states of multiple systems. By choosing to view multiple systems together as single systems, we immediately obtain a specification of how measurements must work for multiple systems — provided that all systems are measured.

For example, if the probabilistic state of two bits (X, Y) is described by the probability vector

$$\frac{1}{2}|00\rangle + \frac{1}{2}|11\rangle,$$

then the outcome 00 — meaning 0 for the measurement of X and 0 for the measurement of Y — is obtained with probability 1/2 and the outcome 11 is also obtained with probability 1/2.

In each case we update the probability vector description of our knowledge accordingly, so that the probabilistic state becomes $|00\rangle$ or $|11\rangle$, respectively.

Partial Measurements : Suppose, however, that we choose not to measure every system, but instead we just measure some proper subset of the systems. This will result in a measurement outcome for each system that gets measured, and will also (in general) affect our knowledge of the remaining systems.

Let us focus on the case of two systems, one of which is measured. The more general situation — in which some proper subset of three or more systems is measured — effectively reduces to the case

of two systems when we view the systems that are measured collectively as if they form one system and the systems that are not measured as if they form a second system.

To be precise, let us suppose (as usual) that X is a system having classical state set Σ, that Y is a system having classical state set Γ, and the two systems together are in some probabilistic state. We will consider what happens when we just measure X and do nothing to Y. The situation where just Y is measured and nothing happens to X is handled symmetrically.

First, we know that the probability to observe a particular classical state $a \in \Sigma$ when just X is measured must be consistent with the probabilities we would obtain under the assumption that Y was also measured. That is, we must have

$$\Pr(\mathsf{X} = a) = \sum_{b \in \Gamma} \Pr((\mathsf{X}, \mathsf{Y}) = (a, b)).$$

This is the formula for the so-called **reduced (or marginal) probabilistic state** of X alone.

This formula makes perfect sense at an intuitive level; something very strange would need to happen for it to be wrong. It would mean that the probabilities for X measurements are influenced simply by whether or not Y is also measured, irrespective of the outcome on Y.

If Y happened to be in a distant location, say, another galaxy, this would allow for faster-than-light signaling, which we reject based on our understanding of physics.

Another way to understand this comes from an interpretation of probability as reflecting a degree of belief about the state of the system.

Since a measurement on Y is taken to simply reveal a preexisting state, a different observer looking at X, unaware of the Y measurement, should not have their probabilities changed.

Given the assumption that only X is measured and Y is not, there may in general still exist uncertainty over the classical state of Y. For this reason, rather than updating our description of the probabilistic state of (X, Y) to $|ab\rangle$ for some selection of $a \in \Sigma$ and $b \in \Gamma$, we must update our description so that this uncertainty about Y is properly reflected.

The following conditional probability formula reflects this uncertainty.

$$\Pr(\mathsf{Y} = b \,|\, \mathsf{X} = a) = \frac{\Pr((\mathsf{X}, \mathsf{Y}) = (a, b))}{\Pr(\mathsf{X} = a)}$$

Here, the expression $\Pr(\mathsf{Y} = b | \mathsf{X} = a)$ denotes the probability that $\mathsf{Y} = b$ conditioned on (or given that) $\mathsf{X} = a$.

It should be noted that the expression above is only defined if $\Pr(\mathsf{X} = a)$ is nonzero, for if

$$\Pr(\mathsf{X} = a) = 0,$$

then we obtain the indeterminate form $\frac{0}{0}$.

This is not a problem, though, because if the probability associated with a is zero, then we'll never observe a as an outcome of a measurement of X, so we don't need to be concerned with this possibility.

To express these formulas in terms of probability vectors, consider a probability vector $|\psi\rangle$ describing the joint state of (X, Y).

$$|\psi\rangle = \sum_{(a,b)\in\Sigma\times\Gamma} p_{ab}|ab\rangle$$

Measuring Xalone yields each possible outcome with probabilities

$$\Pr(\mathsf{X} = a) = \sum_{b\in\Gamma} p_{ab}.$$

Thus, the vector representing the probabilistic state of X alone (i.e., the reduced probabilistic state of X) is given by

$$\sum_{a\in\Sigma}\left(\sum_{c\in\Gamma} p_{ac}\right)|a\rangle.$$

Having obtained a particular outcome $a \in \Sigma$ of the measurement of X, the probabilistic state of Y is updated according to the formula for conditional probabilities, so that it is represented by this probability vector:

$$|\pi_a\rangle = \frac{\sum_{b\in\Gamma} p_{ab}|b\rangle}{\sum_{c\in\Gamma} p_{ac}}.$$

In the event that the measurement of X resulted in the classical state a, we therefore update our description of the probabilistic state of the joint system (X, Y) to $|a\rangle \otimes |\pi_a\rangle$.

One way to think about this definition of $|\pi_a\rangle$ is to see it as a normalization of the vector

$$\sum_{b\in\Gamma} p_{ab}|b\rangle,$$

where we divide by the sum of the entries in this vector to obtain a probability vector.

This normalization effectively accounts for a conditioning on the event that the measurement of X has resulted in the outcome a.

Example :

Let's consider a simple example involving the tossing of two coins, X and Y, each having two possible states: Heads (H) or Tails (T).

Suppose we have the following probabilistic state for the joint system (X, Y):

$$|\psi\rangle = \frac{1}{4}|\mathrm{HH}\rangle + \frac{1}{4}|\mathrm{HT}\rangle + \frac{1}{4}|\mathrm{TH}\rangle + \frac{1}{4}|\mathrm{TT}\rangle$$

This means that X and Y are in a state of equal probability of showing Heads or Tails.

Now, let's say we toss only X. To find the reduced probabilistic state of X alone, we sum over all possible outcomes of Y for each outcome of X:

$$\Pr(\mathsf{X} = \mathrm{H}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$$

$$\Pr(\mathsf{X} = \mathrm{T}) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$$

So, the reduced probabilistic state of X alone is represented by the vector:

$$\frac{1}{2}|\mathrm{H}\rangle + \frac{1}{2}|\mathrm{T}\rangle$$

Now, let's see how the measurement of X affects our knowledge of Y when X is measured, and Y is not. We need to calculate the conditional probability vector $|\pi_a\rangle$ for each outcome $a \in \{\mathrm{H}, \mathrm{T}\}$.

1. For $a = \mathrm{H}$:

$$|\pi_{\mathrm{H}}\rangle = \frac{\frac{1}{4}|\mathrm{HH}\rangle + \frac{1}{4}|\mathrm{TH}\rangle}{\frac{1}{2}} = \frac{1}{2}|\mathrm{HH}\rangle + \frac{1}{2}|\mathrm{TH}\rangle$$

2. For $a = \mathrm{T}$:

$$|\pi_{\mathrm{T}}\rangle = \frac{\frac{1}{4}|\mathrm{HT}\rangle + \frac{1}{4}|\mathrm{TT}\rangle}{\frac{1}{2}} = \frac{1}{2}|\mathrm{HT}\rangle + \frac{1}{2}|\mathrm{TT}\rangle$$

Now, these vectors $|\pi_{\mathrm{H}}\rangle$ and $|\pi_{\mathrm{T}}\rangle$ represent the updated state of Y after the measurement of X has shown Heads or Tails, respectively.

In summary, when tossing only X and leaving Y unmeasured, the reduced state of X is calculated. Then, for each possible outcome of X, the state of Y is updated based on conditional probabilities, reflecting our knowledge of Y given the outcome of the measurement on X.

9.1.5 Operations on probabilistic states :

To conclude this discussion of classical information for multiple systems, we will consider operations on multiple systems in probabilistic states.

Returning to the typical set-up where we have two systems X and Y, let us consider classical operations on the compound system (X, Y). Based on the discussion above, we conclude that any such operation is represented by a **stochastic matrix** whose rows and columns are indexed by the **Cartesian product** $\Sigma \times \Gamma$.

For example, suppose that X and Y are bits, and consider an operation with the following description.

```
If X = 1, then perform a NOT operation on Y;
Otherwise, do nothing.
```

This is a deterministic operation known as a ***controlled-NOT*** operation, where X is the control bit that determines whether or not a **NOT** operation should be applied to the target bit Y.

Here is the matrix representation of this operation:

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

Its action on standard basis states is as follows.

$$\begin{aligned} |\,00\rangle &\rightarrow |\,00\rangle \\ |\,01\rangle &\rightarrow |\,01\rangle \\ |\,10\rangle &\rightarrow |\,11\rangle \\ |\,11\rangle &\rightarrow |\,10\rangle \end{aligned}$$

If we were to exchange the roles of X and Y, taking Y to be the control bit and X to be the target bit, then the matrix representation of the operation would become

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

and its action on standard basis states would be like this:

$$\begin{aligned} |\,00\rangle &\rightarrow |\,00\rangle \\ |\,01\rangle &\rightarrow |\,11\rangle \\ |\,10\rangle &\rightarrow |\,10\rangle \\ |\,11\rangle &\rightarrow |\,01\rangle \end{aligned}$$

Another example is the operation having this description:

```
Perform one of the following two operations,
each with probability 1/2 :

    1. Set Y to be equal to X.

    2. Set X to be equal to Y.
```

The matrix representation of this operation is as follows:

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

The action of this operation on standard basis vectors is as follows:

$$\begin{aligned} |\,00\rangle &\rightarrow |\,00\rangle \\ |\,01\rangle &\rightarrow \frac{1}{2}|00\rangle + \frac{1}{2}\,|\,11\rangle \\ |\,10\rangle &\rightarrow \frac{1}{2}|00\rangle + \frac{1}{2}\,|\,11\rangle \\ |\,11\rangle &\rightarrow |\,11\rangle \end{aligned}$$

In these examples, we are simply viewing two systems together as a single system and proceeding as in the previous lesson.

The same thing can be done for any number of systems. For example, imagine that we have three bits, and we increment the three bits modulo 8 — meaning that we think about the **three** bits as encoding a number between 0 and 7 using binary notation, add 1, and then take the remainder after dividing by 8.

We can write this operation like this:

$$\begin{aligned} &| 001\rangle\langle 000 | + | 010\rangle\langle 001 | + | 011\rangle\langle 010 | + | 100\rangle\langle 011 | \\ &+ | 101\rangle\langle 100 | + | 110\rangle\langle 101 | + | 111\rangle\langle 110 | + | 000\rangle\langle 111 | \end{aligned}$$

We could also write it like this:

$$\sum_{k=0}^{7} |(k+1) mod\ 8\rangle\langle k|$$

assuming we have agreed that a number $j \in 0, 1 \ldots, 7$ inside of a ket refers to that number's three-bit binary encoding. A third option is to express this operation as a matrix.

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

Example :

Let's consider a different example involving the tossing of two coins with a more intricate operation.(***Conditional Swap Operation***)

In this example, the matrix represents a conditional swap operation:

- if at least one coin is in the "Tails" state, the states of the two coins are swapped;
- otherwise, the states remain unchanged.

Probabilistic State:

$$|\psi\rangle = \frac{1}{4}|HH\rangle + \frac{1}{4}|HT\rangle + \frac{1}{4}|TH\rangle + \frac{1}{4}|TT\rangle$$

Matrix Representation of the Conditional Swap Operation:

Let's define an operation that swaps the states of the two coins if at least one of them is in the "Tails" state. Otherwise, it leaves the states unchanged.

Let's break down the process of obtaining the matrix for the conditional swap operation step by step.

We want to represent this operation using a matrix. The rows and columns of the matrix correspond to the possible states of the two coins **(HH, HT, TH, TT)**.

1. **HH State:**
    - No coin is in the "Tails" state, so the state remains unchanged. The first column is [1, 0, 0, 0].
2. **HT State:**
    - Swap because the second coin is in the "Tails" state. The second column is [0, 0, 1, 0].
3. **TH State:**
    - Swap because the first coin is in the "Tails" state. The third column is [0, 1, 0, 0].
4. **TT State:**
    - No coin is in the "Tails" state, so the state remains unchanged. The fourth column is [0, 0, 0, 1].

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

This matrix represents the conditional swap operation. Applying this matrix to the probabilistic state will perform the specified swaps or leave the states unchanged based on the conditions.

This operation is reflected in the probabilistic state, where the probabilities are distributed accordingly. The operation matrix shows how each state transforms under the specified operation.

Action on Standard Basis States:

- $|HH\rangle$ remains unchanged.
- $|HT\rangle$ is swapped to $|TH\rangle$.
- $|TH\rangle$ is swapped to $|HT\rangle$.
- $|TT\rangle$ remains unchanged.

Independent Operations : Now suppose that we have multiple systems and we *independently* perform **separate** operations on the systems.

For example, taking our usual set-up of two systems X and Y having classical state sets Σ and Γ, respectively.

Let us suppose that we perform one operation on X and, completely independently, another operation on Y.

- As we know from the previous topic, these operations are represented by **stochastic matrices** — and to be precise,
- let us say that the operation on X is represented by the matrix M and the operation on Y is represented by the matrix N.

Thus, the rows and columns of M have indices that are placed in correspondence with the elements of Σ and, likewise, the rows and columns of N correspond to the elements of Γ.

A natural question to ask is this: if we view X and Y together as a single, compound system (X,Y),

```
What is the matrix that represents the combined action
of the two operations on this compound system?
```

To answer this question, we must first introduce the tensor product of matrices — which is similar to the tensor product of vectors and is defined analogously.

Tensor products of matrices :

The tensor product $\mathsf{M} \otimes \mathsf{N}$ of the matrices

$$M = \sum_{a,b \in \Sigma} \alpha_{ab} |a\rangle\langle b|$$

and

$$N = \sum_{c,d \in \Gamma} \beta_{cd} |c\rangle\langle d|$$

is the matrix

$$\mathsf{M} \otimes \mathsf{N} = \sum_{a,b \in \Sigma} \sum_{c,d \in \Gamma} \alpha_{ab}\beta_{cd} |ac\rangle\langle bd|$$

Equivalently, M and N is defined by the equation

$$\langle ac|\mathsf{M} \otimes \mathsf{N}|bd\rangle = \langle a|\mathsf{M}|b\rangle\langle c|\mathsf{N}|d\rangle$$

being true for every selection of $a, b \in \Sigma$ and $c, d \in \Gamma$.

An alternative, but equivalent, way to describe $\mathsf{M} \otimes \mathsf{N}$ is that it is the unique matrix that satisfies the equation

$$(\mathsf{M} \otimes \mathsf{N})(|\,\phi\rangle \otimes |\,\psi\rangle) = (\mathsf{M}\,|\,\phi\rangle) \otimes (\mathsf{N}\,|\,\psi\rangle)$$

for every possible choice of vectors $|\,\phi\rangle$ and $|\,\psi\rangle$. Here we are assuming that the indices of $|\,\phi\rangle$ correspond to the elements of Σ and the indices of $|\,\psi\rangle$ correspond to Γ.

Following the convention described previously for ordering the elements of Cartesian products, we can also write the tensor product of two matrices explicitly as follows:

$$\begin{pmatrix} \alpha_{11} & \cdots & \alpha_{1m} \\ \vdots & \ddots & \vdots \\ \alpha_{m1} & \cdots & \alpha_{mm} \end{pmatrix} \otimes \begin{pmatrix} \beta_{11} & \cdots & \beta_{1k} \\ \vdots & \ddots & \vdots \\ \beta_{k1} & \cdots & \beta_{kk} \end{pmatrix}$$

$$= \begin{pmatrix} \alpha_{11}\beta_{11} & \cdots & \alpha_{11}\beta 1k & & \alpha_{1m}\beta 11 & \cdots & \alpha_{1m}\beta_{1k} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ \alpha_{11}\beta_{k1} & \cdots & \alpha_{11}\beta kk & & \alpha_{1m}\beta k1 & \cdots & \alpha_{1m}\beta_{kk} \\ & \vdots & & \ddots & & \vdots & \\ \alpha_{m1}\beta_{11} & \cdots & \alpha_{m1}\beta 1k & & \alpha_{mm}\beta 11 & \cdots & \alpha_{mm}\beta_{1k} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ \alpha_{m1}\beta_{k1} & \cdots & \alpha_{m1}\beta kk & & \alpha_{mm}\beta k1 & \cdots & \alpha_{mm}\beta_{kk} \end{pmatrix}$$

Tensor products of three or more matrices are defined in an analogous way.

If $\mathsf{M}_1, \ldots, \mathsf{M}_n$ are matrices whose indices correspond to classical state sets $\Sigma_1, \ldots, \Sigma_n$, then the tensor product $\mathsf{M}_1 \otimes \ldots \otimes \mathsf{M}_n$ is defined by the condition that

$$\boxed{\langle a_1 \ldots a_n | \mathsf{M}_1 \otimes \ldots \otimes \mathsf{M}_n | b_1 \ldots b_n \rangle \;= \langle a_1 | \mathsf{M}_1 | b_1 \rangle \ldots \langle a_n | \mathsf{M}_n | b_n \rangle}$$

for every choice of classical states $a_1, b_1 \in \Sigma_1, \ldots, a_n, b_n \in \Sigma_n$.

Alternatively, we could also define the tensor product of **three or more matrices** recursively, in terms of tensor products of two matrices, similar to what we observed for vectors.

The tensor product of matrices is sometimes said to be multiplicative because the equation

$$\boxed{(\mathsf{M}_1 \otimes \ldots \otimes \mathsf{M}_n)(\mathsf{N}_1 \otimes \ldots \otimes \mathsf{N}_n) \;= (\mathsf{M}_1\mathsf{N}_1) \otimes \ldots \otimes (\mathsf{M}_n\mathsf{N}_n)}$$

is always true, for any choice of matrices $\mathsf{M}_1, \ldots, \mathsf{M}_n$ and $\mathsf{N}_1, \ldots, \mathsf{N}_n$, provided that the products $\mathsf{M}_1\mathsf{N}_1, \ldots, \mathsf{M}_n\mathsf{N}_n$ make sense.

Example :

Let's use a simpler example to illustrate the concept of the tensor product of matrices.

Consider two matrices:

$$\mathsf{M} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

and

$$\mathsf{N} = \begin{pmatrix} x & y \\ z & w \end{pmatrix}$$

Now, let's compute their tensor product, denoted as $\mathsf{M} \otimes \mathsf{N}$:

$$\mathsf{M} \otimes \mathsf{N} = \begin{pmatrix} a \cdot \mathsf{N} & b \cdot \mathsf{N} \\ c \cdot \mathsf{N} & d \cdot \mathsf{N} \end{pmatrix}$$

Expanding each block in the result, we get:

$$\mathsf{M} \otimes \mathsf{N} = \begin{pmatrix} a \cdot x & a \cdot y & b \cdot x & b \cdot y \\ a \cdot z & a \cdot w & b \cdot z & b \cdot w \\ c \cdot x & c \cdot y & d \cdot x & d \cdot y \\ c \cdot z & c \cdot w & d \cdot z & d \cdot w \end{pmatrix}$$

In a more general form, the tensor product of two matrices M and N is given by:

$$\mathsf{M} \otimes \mathsf{N} = \begin{pmatrix} \alpha_{11} \cdot \beta_{11} & \alpha_{11} \cdot \beta_{12} & \ldots & \alpha_{mn} \cdot \beta_{mk} \\ \alpha_{12} \cdot \beta_{11} & \alpha_{12} \cdot \beta_{12} & \ldots & \alpha_{mn} \cdot \beta_{mk} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{mn} \cdot \beta_{11} & \alpha_{mn} \cdot \beta_{12} & \ldots & \alpha_{mn} \cdot \beta_{mk} \end{pmatrix}$$

This is the result of the tensor product operation, where α_{ij} are elements of M and β_{kl} are elements of N.

Suppose we have two matrices:

1. Matrix M_1 with elements α_{ab}
2. Matrix M_2 with elements β_{cd}

The tensor product $M_1 \otimes M_2$ is then defined by the condition:

$$\langle ac|M_1 \otimes M_2|bd\rangle = \langle a|M_1|b\rangle\langle c|M_2|d\rangle$$

Let's consider a specific example with M_1 being a 2×2 matrix and M_2 being a 3×3 matrix:

$$M_1 = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 0 & 1 & 2 \\ 3 & 4 & 5 \\ 6 & 7 & 8 \end{pmatrix}$$

The resulting tensor product $M_1 \otimes M_2$ would be a 6×6 matrix.

Multiplicative Property of Tensor Product

Let's consider three matrices: A, B, and C. The multiplicative property of tensor product states that

$$(A \otimes B) \cdot (C \otimes D) = (A \cdot C) \otimes (B \cdot D)$$

Here, A, B, C, and D are matrices with appropriate dimensions.

As a simple example, consider:

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}, \quad B = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}, \quad C = \begin{pmatrix} 9 & 10 \\ 11 & 12 \end{pmatrix}, \quad D = \begin{pmatrix} 13 & 14 \\ 15 & 16 \end{pmatrix}$$

Verify that the multiplicative property holds:

$$(A \otimes B) \cdot (C \otimes D) = (A \cdot C) \otimes (B \cdot D)$$

Summary :

To summarize the above discussion, we found that

- if M is a probabilistic operation on X,
- N is a probabilistic operation on Y,
- and the two operations are performed independently,

then the resulting operation on the compound system (X,Y) is the tensor product $\mathsf{M} \otimes \mathsf{N}$.

What we see, both here and for probabilistic states, is that tensor products represent ***independence***:

if we have two systems X and Y that are independently in the probabilistic states $|\phi\rangle$ and $|\pi\rangle$,

- then the compound system (X,Y) is in the probabilistic state $|\phi\rangle \otimes |\pi\rangle$;

And if we independently apply probabilistic operations M and N to the two systems independently,

- then the resulting action on the compound system (X,Y) is described by the operation $\mathsf{M} \otimes \mathsf{N}$.

Let us take a look at an example, which recalls a probabilistic operation on a single bit from the previous chapter:

```
if the classical state of the bit is 0
- it is left alone;
and if the classical state of the bit is 1
- it is flipped to 0 with probability 1/2.
```

As we observed, this operation is represented by the matrix

$$\begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix}$$

If this operation is performed on a bit X, and a **NOT** operation is (independently) performed on a second bit Y, then the joint operation on the compound system (X,Y) has the matrix representation

$$\begin{pmatrix} 1 & \frac{1}{2} \\ 0 & \frac{1}{2} \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \frac{1}{2} \\ 1 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 0 \end{pmatrix}$$

By inspection, we see that this is a **stochastic matrix**.

This will always be the case:

```
the tensor product of two or more stochastic matrices is always stochastic.
```

A common situation that we encounter is one in which one operation is performed on one system and nothing is done to another.

In such a case, exactly the same prescription is followed, noting that **doing nothing is represented by the identity matrix**. For example, resetting the bit X to the 0 state and doing nothing to Y yields the ***probabilistic*** (and in fact deterministic) operation on (X,Y) represented by the matrix

$$\begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

9.2 Quantum Information in multiple systems :

We are now prepared to move on to quantum information in the setting of multiple systems. Much like in the previous lesson on single systems, the mathematical description of quantum information for multiple systems is quite similar to the probabilistic case and makes use of similar concepts and techniques.

9.2.1 Quantum States :

Multiple systems can be viewed collectively as single, compound systems. We have already observed this in the probabilistic setting, and the quantum setting is analogous.

That is, quantum states of multiple systems are represented by column vectors having **complex number entries** and **Euclidean norm equal to 1** — just like quantum states of single systems.

In the multiple system case, the indices of these vectors are placed in correspondence with the **Cartesian product of the classical state** sets associated with each of the individual systems (because that is the classical state set of the compound system).

For instance, if X and Y are qubits, then the classical state set of the pair of qubits (X,Y), viewed collectively as a single system, is the Cartesian product $\{0,1\} \times \{0,1\}$.

By representing pairs of binary values as binary strings of length two, we associate this Cartesian product set with the set **{00,01,10,11}**. The following vectors are therefore all examples of quantum state vectors of the pair (X,Y):

$$\frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{6}}|01\rangle + \frac{i}{\sqrt{6}}|10\rangle + \frac{1}{\sqrt{6}}|11\rangle \ , \frac{3}{5}|00\rangle - \frac{4}{5}|11\rangle \ , \& |01\rangle$$

There are variations on how quantum state vectors of multiple systems are expressed, and we can choose whichever variation suits our preferences. Here are some examples, which are for the first quantum state vector above.

1. We may use the fact that $|\ ab\rangle = |\ a\rangle\ |\ b\rangle$(for any classical states a and b) to instead write

$$\frac{1}{\sqrt{2}}|0\rangle|0\rangle - \frac{1}{\sqrt{6}}|0\rangle|1\rangle + \frac{i}{\sqrt{6}}|1\rangle|0\rangle + \frac{1}{\sqrt{6}}|1\rangle|1\rangle$$

2. We may choose to write the tensor product symbol explicitly like this:

$$\frac{1}{\sqrt{2}}|0\rangle \otimes |0\rangle - \frac{1}{\sqrt{6}}|0\rangle \otimes |1\rangle + \frac{i}{\sqrt{6}}|1\rangle \otimes |0\rangle + \frac{1}{\sqrt{6}}|1\rangle \otimes |1\rangle$$

3. We may subscript the kets to indicate how they correspond to the systems being considered, like this:

$$\frac{1}{\sqrt{2}}|0\rangle_X|0\rangle_Y - \frac{1}{\sqrt{6}}|0\rangle_X|1\rangle_Y + \frac{i}{\sqrt{6}}|1\rangle_X|0\rangle_Y + \frac{1}{\sqrt{6}}|1\rangle_X|1\rangle_Y$$

Of course, we may also write quantum state vectors explicitly as column vectors:

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{6}} \\ \frac{i}{\sqrt{6}} \\ \frac{1}{\sqrt{6}} \end{pmatrix}$$

Note : Depending upon the context in which it appears, one of these variations may be preferred — but they are all equivalent in the sense that they describe the same vector.

Tensor products of quantum state vectors : Similar to what we have for probability vectors, tensor products of quantum state vectors are also quantum state vectors — and again they represent independence among systems.

In greater detail, and beginning with the case of two systems, suppose that

- $|\,\phi\rangle$ is a quantum state vector of a system X and
- $|\psi\rangle$ is a quantum state vector of a system Y.
- The tensor product $|\phi\rangle \otimes |\psi\rangle$, which may alternatively be written as $|\phi\rangle|\psi\rangle$ or as $|\phi \otimes \psi\rangle$, is then a **quantum state vector** of the **joint system** (X,Y).

We refer to a state of this form as a being a ***product state***.

Intuitively speaking, when a pair of systems (X,Y) is in a product state $|\phi\rangle \otimes |\psi\rangle$, we may interpret this as meaning that X is in the quantum state $|\,\phi\rangle$, Y is in the quantum state $|\psi\rangle$, and the states of the two systems have nothing to do with one another.

The fact that the tensor product vector $|\phi\rangle \otimes |\psi\rangle$ is indeed a quantum state vector is consistent with the **Euclidean norm** being multiplicative with respect to tensor products:

$$\boxed{\begin{aligned}
\big|\big||\phi\rangle \otimes |\psi\rangle\big|\big| &= \sqrt{\sum_{(a,b)\in\Sigma\times\Gamma} |\langle ab|\phi\otimes\psi|^2} \\
&= \sqrt{\sum_{a\in\Sigma}\sum_{b\in\Gamma} |\langle a|\phi\rangle\langle b|\psi\rangle|^2} \\
&= \sqrt{\Big(\sum_{a\in\Sigma} |\langle a|\phi\rangle|^2\Big)\Big(\sum_{b\in\Gamma} |\langle b|\psi\rangle|^2\Big)} \\
&= \big|\big||\phi\rangle\big|\big|\,\big|\big||\psi\rangle\big|\big|
\end{aligned}}$$

Thus, because $|\,\phi\rangle$ and $|\psi\rangle$ are quantum state vectors, we have $|\,\phi\rangle = 1$ and $|\,\psi\rangle = 1$ and therefore $|\phi\rangle \otimes |\psi\rangle = 1$, so $|\phi\rangle \otimes |\psi\rangle$ is also a ***quantum state vector***.

This discussion may be generalized to more than two systems. If $|\psi_1\rangle, \ldots, |\psi_n\rangle$ are quantum state vectors of systems $\mathsf{X}_1, \ldots, \mathsf{X}_n$, then $|\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle$ is a quantum state vector representing a *product state* of the joint system $(\mathsf{X}_1, \ldots, \mathsf{X}_n)$. Again, we know that this is a quantum state vector because

$$\big|\big||\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle\big|\big| = \big|\big||\psi_1\rangle\big|\big| \ldots \big|\big||\psi_n\rangle\big|\big| = 1^n = 1$$

9.2.2 Quantum Entanglement :

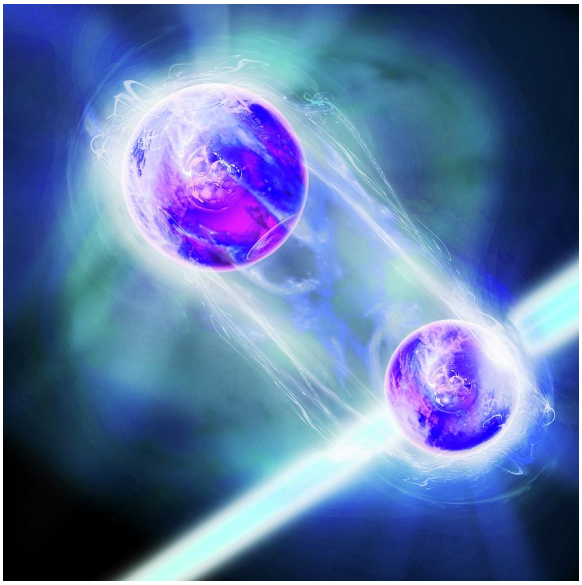
```
Quantum entanglement is a phenomenon that occurs when two or more
quantum particles become deeply interconnected in such a way that the
state of one particle is intrinsically linked to the state of another,
regardless of the distance between them.
```

This connection persists even if the particles are separated by vast distances, which famously led Einstein to refer to it as **"spooky action at a distance"**.

Now, when it comes to qubits, entanglement plays a pivotal role in the power and potential of quantum computing. Here's how:

1. **Entangled Qubits**:
    - In a quantum system, when qubits become entangled, the state of one qubit becomes dependent on the state of the other(s), and vice versa. This connection exists irrespective of the physical separation between the qubits.
    - For instance, if two qubits become entangled, measuring the state of one qubit instantly determines the state of the other, regardless of the distance between them. This instantaneous correlation is a hallmark of entanglement.
2. **Effect on Qubits**:
    - Entanglement allows for the creation of highly correlated quantum states between qubits. This correlation persists even if the entangled qubits are separated by large distances.
    - When qubits are entangled, they share a joint quantum state, and measurements on one qubit will instantaneously affect the state of the other qubit, ensuring they remain correlated.
3. **Applications in Quantum Computing**:
    - Entanglement is crucial in quantum computing algorithms. It enables the manipulation of qubits in a way that exploits their interconnectedness to perform computations that classical computers find extremely challenging or impossible to achieve efficiently.
    - Quantum algorithms, such as Shor's algorithm for factoring large numbers or quantum teleportation, leverage entanglement to perform computations more efficiently than classical counterparts.
4. **Quantum Communication**:
    - Entanglement also has implications for quantum communication and cryptography. It allows for secure communication protocols, such as quantum key distribution (QKD), where entangled qubits are used to securely transmit encryption keys.
5. **Challenges**:
    - While entanglement is a powerful resource in quantum computing, it's delicate and easily disrupted by interactions with the environment, causing a phenomenon known as decoherence. Efforts in quantum computing research focus on methods to preserve and harness entanglement for practical computational purposes.

In essence, entanglement is a remarkable feature of quantum physics that underlies the potential of quantum computing. It enables the creation of highly correlated quantum states between qubits, offering computational advantages and opening up possibilities for secure communication and cryptography in the quantum realm.



Entangled States :

1. **Definition of a Product State:** A quantum state vector $|\phi\rangle$ is said to be a product state if it can be expressed as the tensor product of state vectors $|\phi_1\rangle$ and $|\phi_2\rangle$ for subsystems:

$$|\phi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle.$$

2. **Entanglement as Non-Product State:**

   Not all quantum state vectors of multiple systems are product states. For example, the quantum state vector

$$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle \tag{a}$$

   of two qubits is **not a product state**.

   The given quantum state vector

$$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

   is considered to be entangled because it cannot be expressed as a product state.

   That is, if (a) was a product state, there would exist quantum state vectors $|\phi\rangle$ and $|\psi\rangle$ such that

$$|\phi\rangle \otimes |\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle.$$

   But then it would necessarily be the case that

$$\langle 0 \mid \phi\rangle\langle 1 \mid \psi\rangle = \langle 01 \mid \phi \otimes \psi\rangle \ = 0$$

implying that $\langle 0 \mid \phi \rangle = 0$ or $\langle 1 \mid \psi \rangle = 0$ (or boths). That contradicts the fact that

$$\boxed{\begin{aligned} \langle 0 \mid \phi \rangle \langle 0 \mid \phi \rangle &= \langle 00 | \phi \otimes \psi \rangle = \frac{1}{\sqrt{2}} \\ & \& \\ \langle 1 \mid \phi \rangle \langle 1 \mid \phi \rangle &= \langle 11 | \phi \otimes \psi \rangle = \frac{1}{\sqrt{2}} \end{aligned}}$$

are both **non zero**.

3. **Generalization to Any Non-Product State:**

   Notice that the specific value $1/\sqrt{2}$ is not important to this argument — what is important is that this value is **nonzero**.

   It's emphasized that the specific coefficients are not crucial; any nonzero coefficients would lead to the same conclusion. For instance, the state $\frac{3}{5}|00\rangle + \frac{4}{5}|11\rangle$ is also considered entangled.

4. **Correlation and Entanglement:**

   The text then equates entanglement to correlation, at least in the simplified formulation being discussed. Any state vector that is not a product vector is considered entangled.

   Entanglement is a quintessential feature of quantum information that will be discussed in much greater detail in later lessons. Entanglement can be complicated, particularly for the sorts of noisy quantum states that can be described in the general, density matrix formulation of quantum information that was mentioned in previous concepts — but for quantum state vectors in the simplified formulation that we are focusing on in this unit, **entanglement** is equivalent to **correlation**.

   That is, **Any quantum state vector that is not a product vector represents an entangled state**.

5. **Example of a Product State:**
   In contrast, the quantum state vector

   $$\frac{1}{2}|00\rangle + \frac{i}{2}|01\rangle - \frac{1}{2}|10\rangle - \frac{i}{2}|11\rangle$$

   is an example of product state as it can be factored into two independent state vectors:

   $$\frac{1}{2}|00\rangle + \frac{i}{2}|01\rangle - \frac{1}{2}|10\rangle - \frac{i}{2}|11\rangle = \left(\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle\right) \otimes \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle\right)$$

   Hence this state is not entangled.

In essence, entanglement is identified with the **inability to express a state vector as a product of state vectors for individual subsystems**, and this concept is illustrated with specific examples.

9.2.3 Bell States :

Bell states, also known as **EPR pairs** (after **Einstein**, **Podolsky**, and **Rosen**), are specific entangled quantum states of two qubits. There are four Bell states, and they play a crucial role in quantum information and quantum entanglement. Here are some key notes about Bell states:

1. **Definition of Bell States:** The general form of a Bell state for two qubits, written in the computational basis $|00\rangle, |01\rangle, |10\rangle, |11\rangle$, is given by:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),$$
$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$$
$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle),$$
$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

2. **Entanglement:** Bell states are maximally entangled. This means that the measurement outcome of one qubit instantaneously determines the state of the other, regardless of the physical separation between them.

3. **Use in Quantum Information:** Bell states are often used in quantum information protocols, including quantum teleportation and quantum key distribution (QKD). They form a basis for quantum communication tasks that leverage the principles of entanglement.

4. **Quantum Teleportation:** The concept of quantum teleportation relies on the use of Bell states. In this protocol, an unknown quantum state can be transmitted from one location to another using two entangled particles and classical communication.

5. **Quantum Key Distribution (QKD):** Bell states are employed in quantum key distribution protocols, where two parties can establish a secret cryptographic key with the guarantee of security against eavesdropping. The security of QKD is based on the principles of quantum entanglement.

6. **Measurement Correlations:** Measurement correlations in Bell states exhibit quantum correlations that violate classical correlations predicted by local realism. This phenomenon, known as **Bell inequality violation**, has been experimentally confirmed and is a cornerstone in understanding the non-classical nature of quantum mechanics.

7. **Relation to Bell's Theorem:** Bell states are related to Bell's theorem, which shows that no local hidden variable theory can reproduce all the predictions of quantum mechanics. Bell's theorem and the associated Bell inequalities provide a testable way to experimentally verify the non-local correlations predicted by quantum mechanics.

8. **Quantum Gates and Quantum Circuits:** Bell states are used in the construction of quantum gates and circuits. For example, the Controlled-NOT (CNOT) gate applied to a Bell state produces an entangled state, illustrating the use of Bell states in quantum computation.

9. **Superposition of Bell States:** It's possible to create superpositions of Bell states, leading to more complex quantum states with enhanced quantum properties. Superpositions of Bell states are used in various quantum algorithms and protocols.

In summary, Bell states are fundamental in quantum information science, playing a central role in tasks like quantum communication, teleportation, and quantum key distribution. They showcase the unique and non-local correlations inherent in quantum mechanics.

Bells Theorem :

1. **Introduction:**
   - Bell's Theorem by physicist John Bell explores the strange behavior of entangled particles, challenging our usual understanding of how things work.
2. **Bell's Inequalities:**
   - Mathematical checks to see if the peculiar quantum correlations violate what classical physics predicts.
3. **Quantum Entanglement:**
   - Entangled particles are like magic twins; what happens to one instantly affects the other, no matter how far apart.
4. **Coin Example:**
   - Imagine two magical coins - "Heads" (H) for spin-up, "Tails" (T) for spin-down.
5. **Classical Predictions:**
   - In regular physics, coin outcomes are independent. We expect equal chances for "HT," "TH," "HH," and "TT."

$$P(H = H) + P(H = T) + P(T = H) + P(T = T) = 1$$

6. **Quantum Predictions:**
   - Quantum magic predicts a special connection between the coins. If one shows "H," the other likely shows "T," and vice versa. This correlation is different from what we'd expect in regular physics.
7. **Bell's Inequalities Test:**
   - Bell's rules let us compare what we see with what regular physics expects. If our magical coins consistently break those rules, it hints at quantum weirdness.
8. **Experimental Results:**
   - Testing our magic coins repeatedly, we observe more "HT" and "TH" outcomes than "HH" and "TT." This tendency consistently violates classical predictions.

$$P(T = H) + P(H = T) \geq 1$$

   - Even though we didn't know about the magic beforehand, the coins' consistent rule-breaking strongly suggests something beyond regular physics is at play.
9. **Inference:**

- The breaking of expected rules in our experiments hints at a mysterious quantum connection between the coins, suggesting the influence of quantum entanglement.

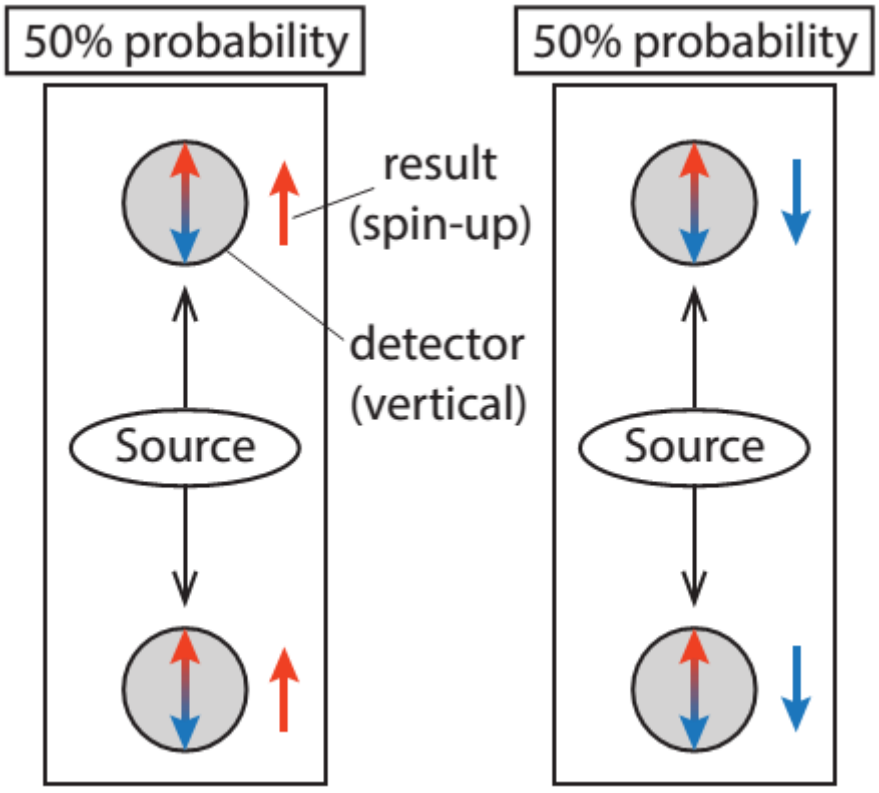
11. **Conclusion:**
    - Bell's Theorem takes us on a journey into the puzzling world of quantum entanglement, where coins act like magic buddies, consistently defying the rules of classical physics.

Example : Although the predictions of quantum theory are well-understood, its interpretation is famously difficult. Because quantum mechanics only makes probabilistic predictions, many people have desired a **"hidden variable"** interpretation, where quantum objects have definite states, despite appearances to the contrary. But hidden variable interpretations are generally not accepted.

It is certainly possible to create a hidden variable interpretation that agrees with the all the predictions of quantum theory. However, they have a number of unsatisfying properties. Indeed there are a few theorems that prove that any hidden variable interpretation must have **unsatisfying** properties.

The most important of these is Bell's Theorem, formulated in 1964.

- Bell's Theorem considers a particular thought experiment, in which two electrons are emitted simultaneously from a single source in opposite directions.
- This source emits electrons that are **"entangled"**, meaning that their quantum states are correlated with one another.
- If we perform the same measurement on both electrons, both measurements will always produce the same result.



(***Fig. 1:*** *The electrons are entangled such that when you measure vertical spin of both, either both electrons are spin up, or both are spin down.*)

The particular thing we want to measure is the electron's spin. Electron spin is quantized; if we measure the spin in the vertical direction, we will either measure **+1/2** or **-1/2** (in the appropriate units), and never anything between.

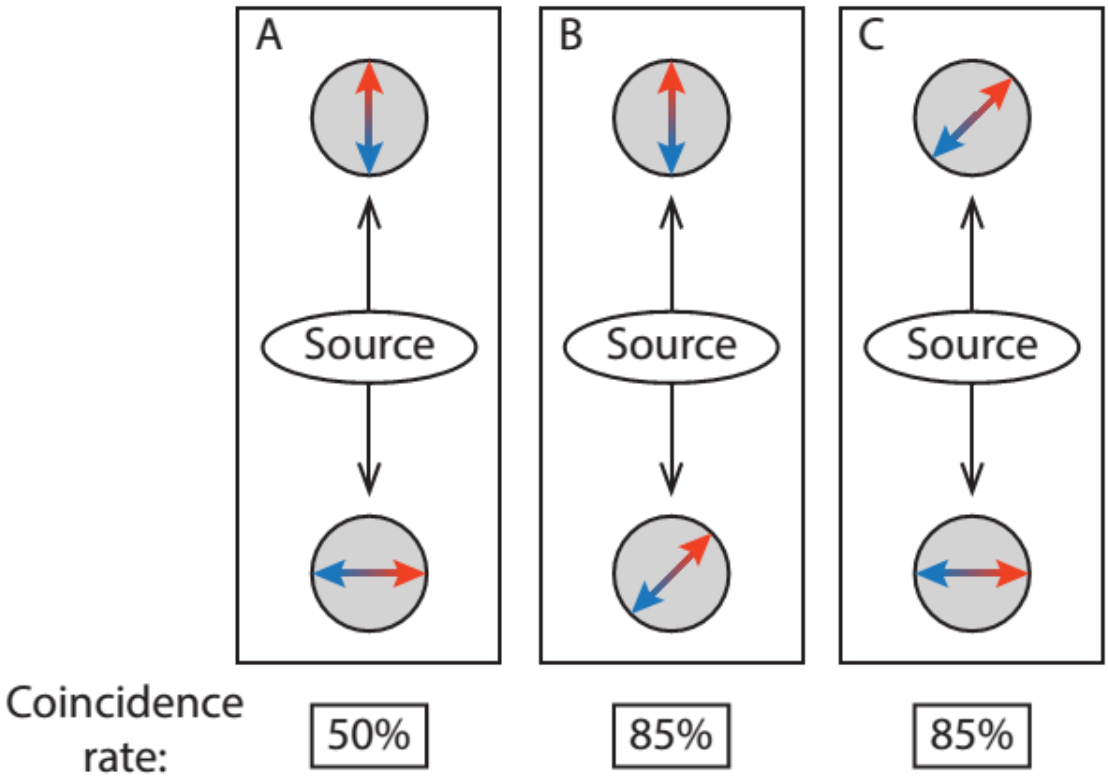
If the result is **+1/2**, we call it **spin-up**, and if it's **-1/2**, we call it **spin down**. We can also measure the spin in the horizontal direction, and again we will always measure +1/2 or -1/2.

Now, we have heard that in quantum mechanics, it is **impossible** to know both the *position* and *momentum* of a particle. This is a common occurrence in quantum theory, when it is impossible to make two measurements at the same time.

- This is true of **horizontal spin** and **vertical spin**. We can measure the vertical spin of an electron, and we can measure the horizontal spin, but **we can't measure both at the same time**.

While we cannot measure both the horizontal and vertical spin of a single electron, we can do the next best thing, and **measure two entangled electrons simultaneously**.

Preferably these measurements are performed sufficiently far apart that no information can be exchanged between electrons without traveling faster than light. And furthermore, we don't need to restrict ourselves to horizontal and vertical directions, we can make our detectors freely rotate to any angle. In particular, we consider three configurations, with angles **0 degrees, 45 degrees, and 90 degrees** from **vertical**.



(***Fig. 2:*** *We measure the spin of the two electrons simultaneously.*

Experiment A: We measure one at 0 degrees, and one at 90 degrees.

Experiment B: We measure one at 0 degrees, and one at 45 degrees.

Experiment C: We measure one at 45 degrees, and one at 90 degrees.

The "coincidence rate" is the probability that both detectors will measure the same spin (in the direction measured, ie both red, or both blue). Quantum theory predicts a coincidence rate of $(1 + cos(x))/2$, where x is the relative angle between measurements.)

The probability paradox

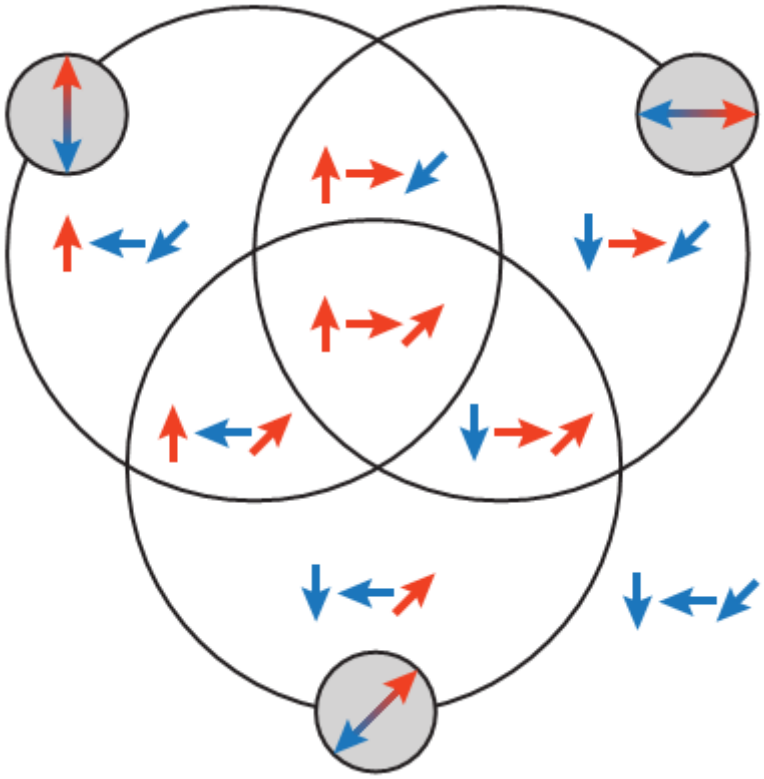
> `Under a hidden variable interpretation, each electron has a definite state even if it's something we never measure.`

Every electron is either **spin-up or spin-down**, and every electron is either **spin-left or spin-right**.

- For Bell's Theorem, we also need to consider the **spin along a 45 degree diagonal**; every electron is either **spin-up-right or spin-down-left**.
- Putting it all together, every electron is in one of the following eight states:

$$(up, right, up - right)$$
$$(up, right, down - left)$$
$$(up, left, up - right)$$
$$(up, left, down - left)$$
$$(down, right, up - right)$$
$$(down, right, down - left)$$
$$(down, left, up - right)$$
$$(down, left, down - left)$$

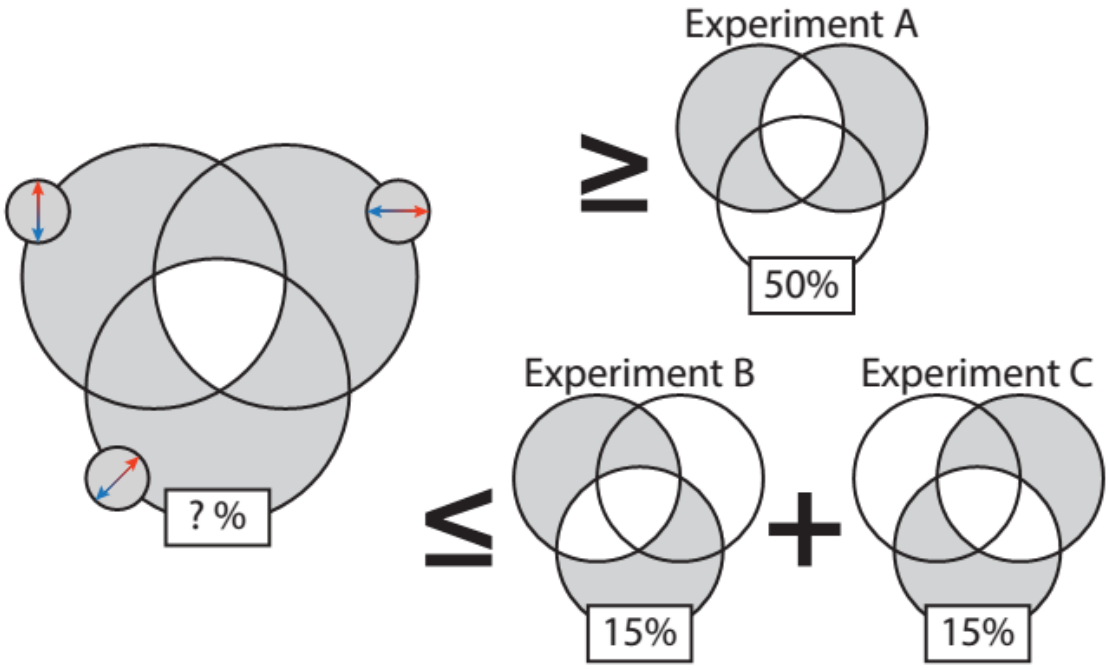
These eight possible states can be summarized in the Venn diagram below.



(***Fig. 3:*** *A Venn diagram illustrating eight possible hidden variable states that an electron may have in the Bell's Theorem thought experiment.*)

If you measure both electrons in the same direction, then they produce identical results. This indicates that both electrons are in the same section of the Venn diagram, although we don't know exactly which section that is. In fact, we don't even know what the probabilities are for each section. Since we can only measure two of the circles of the Venn diagram at a time, we will have to do some sleuthing to find out.

However, if we go through the sleuthing, we will find that the *probabilities don't make any sense!* This can be illustrated in a single picture (Fig. 4 below), using the coincidence rates predicted by quantum theory (shown in Fig. 2).



(***Fig. 4:*** *An illustration of bell's inequality. We consider the probability that the electrons are in the shaded region of the leftmost Venn diagram. It must be at least as large as the shaded region in the upper right diagram, which we determined was 50% in experiment A (see Fig. 2). Additionally, it must be no more than the sum of the shaded regions in the lower left diagrams, which we determined are 15% each, using experiments B and C.*)

Figure 4 shows a contradiction. The shaded region of the leftmost Venn diagram must be at least 50%, but also no more than 30%. **It cannot be both!**

Bell States (contd.): We will now take a look as some important examples of multiple-qubit quantum states, beginning with the *Bell states*. These are the following four two-qubit states:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle),$$
$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle),$$
$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle),$$
$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle).$$

Notice that the same argument that establishes that $|\Phi^+\rangle$ is **not a product state** reveals that **none of the other Bell states is a product state either** — ***all four of the Bell states represent entanglement between two qubits***.

The collection of all four Bell states $\{|\Phi^+\rangle, |\Phi^-\rangle, |\Psi^+\rangle, |\Psi^-\rangle\}$ is known as ***Bell Basis***;

```
Any quantum state vector of two qubits, or indeed any complex vector at
all having entries corresponding to the four classical states of two
bits, can be expressed as a linear combination of the four Bell states.
```

For example,

$$|00\rangle \ = \frac{1}{\sqrt{2}}|\phi^+\rangle \ + \frac{1}{\sqrt{2}}|\phi^-\rangle$$

GHZ and W States : Next we will consider two interesting examples of states of three qubits.

The first example, which we will consider represents a quantum of three qubits $(\mathsf{X}, \mathsf{Y}, \mathsf{Z})$, is the **GHZ** state (so named in honor of *Daniel Greenberger*, *Michael Horne*, and *Anton Zeilinger*, who first studied some of its properties):

$$\boxed{\frac{1}{\sqrt{2}}|000\rangle \ + \frac{1}{\sqrt{2}}|111\rangle}$$

The second example is the so-called **W-state** :

$$\boxed{\frac{1}{\sqrt{3}}|001\rangle \ + \frac{1}{\sqrt{3}}|010\rangle \ + \frac{1}{\sqrt{3}}|100\rangle}$$

Neither of these states is a product state, meaning that they cannot be written as a tensor product of three qubit quantum state vectors.

We will examine both of these two states further when we discuss partial measurements of quantum states of multiple systems.

Additional Example :

The examples of quantum states of multiple systems we have seen so far are states of two or three qubits, but we can also have quantum states of multiple systems having different classical state sets.

For example, here is a quantum state of three systems, X,Y and Z, where the classical state set of X is the binary alphabet (so X is a **qubit**) and the classical state set of Y and Zis $\{\clubsuit, \diamondsuit, \heartsuit, \spadesuit\}$:

$$\frac{1}{2}|0\rangle|\heartsuit\rangle|\heartsuit\rangle \ + \frac{1}{2}|1\rangle|\spadesuit\rangle|\heartsuit\rangle \ - \frac{1}{\sqrt{2}}|0\rangle|\heartsuit\rangle|\diamondsuit\rangle$$

And, here is an example of a quantum state of three systems (X,Y,Z) where X,Y and Z all share the same classical state set $\{0, 1, 2\}$:

$$\frac{|012\rangle - |021\rangle + |120\rangle - |102\rangle + |201\rangle - |210\rangle}{\sqrt{6}}$$

Systems having the classical state set $\{0, 1, 2\}$ are often called ***trits*** or, assuming we consider the possibility that they are in quantum states, ***qutrits***.

The term qudit refers to a system having classical state set $\{0, \ldots, d-1\}$ for an arbitrary choice of d.

9.2.4 Measurements of Quantum States :

Standard basis measurements of quantum states of single systems were discussed in the previous topics: if a system having classical state set Σ is in a quantum state represented by the vector $|\ \psi\rangle$, and that system is measured (*with respect to a standard basis measurement*), then each classical state $a \in \Sigma$ appears with probability $\mathbf{|\langle a|\ \rangle|^2}$.

This tells us what happens when we have a quantum state of multiple systems and choose to measure the entire compound system (which is equivalent to measuring all of the systems).

To state this precisely, let us suppose that $\mathsf{X}_1, \ldots, \mathsf{X}_N$ are systems having classical state sets $\Sigma_1, \ldots, \Sigma_N$ respectively.

We may then view $(\mathsf{X}_1, \ldots, \mathsf{X}_N)$ collectively as a single system whose classical state set is the Cartesian product $\Sigma_1 \times \ldots \times \Sigma_N$.

If a quantum state of this system is represented by the quantum state vector $|\ \psi\rangle$, and all of the systems are measured, then each possible outcome $(a_1, \ldots, a_n) \in \Sigma_1 \times \ldots \times \Sigma_N$ appears with probability $|\langle a_1 \ldots a_n|\psi\rangle|^2$.

For example, if systems X and Y are jointly in the quantum state

$$\frac{3}{5}|0\rangle|\heartsuit\rangle - \frac{4i}{5}|1\rangle|\spadesuit\rangle,$$

then measuring both systems with respect to a standard basis measurement yields the outcome $(0, \heartsuit)$ with probability $9/25$ and the outcome $(1, \spadesuit)$ with probability $16/25$.

Partial measurements for two systems : Now let us consider the situation in which we have multiple systems in some quantum state, and we measure a proper subset of the systems. As before, we will begin with two systems X and Y having classical state sets Σ and Γ, respectively.

In general, a quantum state vector of (X,Y) takes the form

$$|\psi\rangle\ = \sum_{(a,b)\in\Sigma\times\Gamma} \alpha_{ab}|ab\rangle,$$

where $\{\alpha_{ab} : (a,b) \in \Sigma \times \Gamma\}$ is a collection of complex numbers satisfying

$$\sum_{(a,b)\in\Sigma\times\Gamma} |\alpha_{ab}|^2\ = 1$$

(which is equivalent to $|\psi\rangle$ being **unit vector**).

We already know, from the discussion above, that if both X and Ywere measured, then each possible outcome $(a,b) \in \Sigma \times \Gamma$ would appear with probability

$$|\langle ab|\psi\rangle|^2\ = |\alpha_{ab}|^2$$

Supposing that just the first system X is measured, the probability for each outcome $a \in \Sigma$ to appear must therefore be equal to

$$Pr(X\ = a)\ = \sum_{b\in\Gamma} |\langle ab|\psi\rangle|^2\ = \sum_{b\in\Gamma} |\alpha_{ab}|^2$$

This is consistent with what we already saw in the probabilistic setting, and is once again consistent with our understanding of physics. That is, the probability for each particular outcome to appear when X is measured cannot possibly depend on whether or not Y was also measured, as that would otherwise allow for faster-than-light communication.

Having obtained a particular outcome $a \in \Sigma$ of this measurement of X, we expect that the quantum state of X changes so that it is equal to $|\, a\rangle$, like we had for single systems.

But what happens to the quantum state of Y ?

To answer this question, let us describe the joint quantum state of (X,Y) under the assumption that X **was measured** (with respect to a standard basis measurement) and the result was the **classical state** a.

First we express the vector $|\psi\rangle$ as

$$|\psi\rangle \ = \sum_{a \in \Sigma} |a\rangle \otimes |\phi_a\rangle,$$

where

$$|\phi_a\rangle \ = \sum_{b \in \Gamma} \alpha_{ab} |b\rangle$$

for each $a \in \Sigma$.

Notice that the probability that the standard basis measurement of X results in each outcome a may be written as follows:

$$\sum_{b \in \Gamma} |\alpha_{ab}|^2 \ = \left\| |\phi_a\rangle \right\|^2$$

Now, as a result of the standard basis measurement of X resulting in the outcome a, we have that the quantum state of the pair (X,Y) together becomes

$$|a\rangle \otimes \frac{|\phi_a\rangle}{\left\| |\phi_a\rangle \right\|}$$

That is, the state "collapses" like in the single-system case, but only as far as is required for the state to be consistent with the measurement of X having produced the outcome a.

Informally speaking, $|a\rangle \otimes \phi_a\rangle$ represents the component of $|\psi\rangle$ that is consistent with the a measurement of X resulting in the outcome a. We normalize this vector — by dividing it by its **Euclidean norm**, which is equal to $\left\| |\phi_a\rangle \right\|$ — to yield a valid quantum state vector having Euclidean norm equal to 1. This **normalization** step is analogous to what we did in the probabilistic setting when we divided vectors by the sum of their entries to obtain a *probability vector*.

Example :

Let's use the example of two entangled coins, X and Y, where each coin can land on either "Heads" (H) or "Tails" (T). The quantum state vector $|\psi\rangle$ for the joint system (X, Y) can be expressed as:

$$|\psi\rangle = \alpha_{HH}|HH\rangle + \alpha_{HT}|HT\rangle + \alpha_{TH}|TH\rangle + \alpha_{TT}|TT\rangle$$

Here, $|ab\rangle$ represents the state where X is in state a (H or T) and Y is in state b (H or T). The coefficients α_{ab} are complex numbers, and the sum of their squared magnitudes is equal to 1, ensuring that $|\psi\rangle$ is a unit vector.

Now, let's consider the scenario where we measure system X (the first coin) and get the outcome "Heads" (H).

The probability of this event, denoted as $P(X = H)$, is calculated by summing the squared magnitudes of the probability amplitudes associated with the outcomes where the first coin is "Heads."

The probability $P(X = H)$ is calculated as follows:

$$P(X = H) = |\alpha_{HH}|^2 + |\alpha_{HT}|^2$$

This represents the probability of measuring "Heads" on the first coin regardless of the outcome of the second coin.

After this measurement, the quantum state of system X collapses to $|H\rangle$, and the joint state becomes:

$$|H\rangle \otimes \frac{|\phi_H\rangle}{||\phi_H\rangle||}$$

Here, $|\phi_H\rangle$ represents the state of system Y given that X is in state "Heads." The probability amplitudes $\alpha_{H\star}$ are used to construct this state.

Let's see a simple example:

$$|\psi\rangle = \frac{1}{\sqrt{2}}|HH\rangle + \frac{1}{\sqrt{2}}|TT\rangle$$

In the example where the state vector is $|\psi\rangle = \frac{1}{\sqrt{2}}|HH\rangle + \frac{1}{\sqrt{2}}|TT\rangle$, the probability $P(X = H)$ is:

$$P(X = H) = \left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2}$$

This indicates that, in this particular state, the probability of measuring "Heads" on the first coin is 1/2. The outcome of the first coin is correlated with the outcome of the second coin in such a way that the probability distribution is uniform, reflecting the maximally entangled nature of the state.

In this case, after measuring system X and obtaining the outcome "Heads," the state of the combined system becomes:

$$|H\rangle \otimes \frac{|\phi_H\rangle}{||\phi_H\rangle||} = \frac{1}{\sqrt{2}}|H\rangle \otimes \frac{|H\rangle}{\sqrt{2}} + \frac{1}{\sqrt{2}}|H\rangle \otimes \frac{|T\rangle}{\sqrt{2}}$$

This illustrates the concept of state collapse and normalization in the context of entangled coins. The state $|H\rangle$ is now the state of system X (the first coin), and the state of system Y (the second coin) is a normalized version of its original state given that X is "Heads."

This example reflects the quantum principles of entanglement and the impact of measurements on the joint quantum state of entangled systems.

In the context of quantum mechanics and the tensor product formalism, the expression $|H\rangle \otimes \frac{|\phi_H\rangle}{\|\phi_H\rangle\|}$ represents the quantum state of system Y given that system X is in the state "Heads" ($|H\rangle$).

Here's a breakdown of the components:

- $|H\rangle$ is the quantum state of system X (the first coin) after the measurement outcome "Heads."
- $|\phi_H\rangle$ is the quantum state of system Y (the second coin) given that system X is in the state "Heads."
- $\|\phi_H\rangle\|$ is the Euclidean norm or the magnitude of $|\phi_H\rangle$, ensuring that the resulting state is normalized (having a magnitude of 1).

So, $|H\rangle \otimes \frac{|\phi_H\rangle}{\|\phi_H\rangle\|}$ is the joint quantum state of systems X and Y, where X is in the state "Heads," and Y is in the state $|\phi_H\rangle$ normalized to have a magnitude of 1.

This is a fundamental concept in quantum mechanics, where measurements can cause the quantum state to collapse, and the resulting joint state reflects the correlation between the measured outcomes of entangled systems.

Example when we consider 3 systems :

If we have three entangled coins, labeled X, Y, and Z, the joint quantum state of the system would be described by a state vector involving all possible combinations of outcomes for the three coins. Let's denote the states "Heads" (H) and "Tails" (T) for each coin as $|H\rangle$ and $|T\rangle$, respectively.

The general form of the joint quantum state for three coins (X, Y, and Z) can be expressed as:

$$|\psi\rangle = \sum_{a,b,c \in \{H,T\}} \alpha_{abc} |abc\rangle$$

Here, α_{abc} are complex numbers, and the sum of the squared magnitudes of these coefficients is equal to 1 to ensure that $|\psi\rangle$ is a normalized state vector.

Suppose you measure the first coin (X) and get the outcome "Heads" (H). After this measurement, the quantum state of the system would collapse, and the joint state becomes:

$$|H\rangle \otimes \frac{|\phi_H\rangle}{\|\phi_H\rangle\|} \otimes \frac{|\chi_H\rangle}{\|\chi_H\rangle\|}$$

Here, $|\phi_H\rangle$ represents the state of system Y given that X is in the state "Heads," and $|\chi_H\rangle$ represents the state of system Z given that X is in the state "Heads." Both states are normalized to have magnitudes of 1.

The general form of the joint state after measuring X and obtaining "Heads" would be:

$$|H\rangle \otimes \frac{|\phi_H\rangle}{\|\phi_H\rangle\|} \otimes \frac{|\chi_H\rangle}{\|\chi_H\rangle\|}$$

Similarly, if we measure the second coin (Y) or the third coin (Z), the joint state would collapse accordingly, and the corresponding subsystem states would be normalized based on the outcomes of the measurements.

This extension to three coins demonstrates the entanglement between multiple quantum systems and how measurements on one system can influence the states of the other entangled systems. The principles discussed earlier for two coins can be generalized to a larger number of entangled coins.

Limitation of Simplified description of Quantum Information : In classical probability, we often deal with joint probability distributions and marginal probabilities. For example, if we have two independent coins, X and Y, and we know the joint probability distribution of their outcomes, we can easily calculate the probability of each coin individually. The probability distribution of X is a marginal probability obtained by summing or integrating over all possible outcomes of Y, and vice versa.

However, in quantum mechanics, the concept of a reduced or marginal state is more complex.

- This is because the quantum state of a composite system, such as two entangled coins, cannot always be expressed as independent states for each subsystem.
- In other words, we can't always describe the state of one coin independently of the other, even if they are entangled.

Let's consider the entangled state $|\psi\rangle = \frac{1}{\sqrt{2}}|HH\rangle + \frac{1}{\sqrt{2}}|TT\rangle$ for two entangled coins.

- In classical terms, this state might suggest that each coin is in a superposition of states: $|H\rangle$ and $|T\rangle$.
- However, the true nature of entanglement is such that the overall state $|\psi\rangle$ doesn't break down into independent states for each coin.

If we try to describe the state of just the first coin, we can't simply say it's in $|H\rangle$ with a certain probability and $|T\rangle$ with another probability, as we would in classical probability.

The entangled nature of the state means that the outcomes of one measurement are correlated with the outcomes of the other.

In other words, the entangled state $|\psi\rangle$ doesn't easily allow us to describe the reduced state of just the first coin or just the second coin independently.

- The correlation between the coins is an **inherent feature** of the entangled state and is not captured by independent states for each coin.

This limitation in describing reduced or marginal states independently is one of the unique features of quantum entanglement, distinguishing it from classical probability.

Specifically, for a probabilistic state of two systems (X,Y) described by a probability vector

$$|\psi\rangle \ = \sum_{(a,b)\in\Sigma\times\Gamma} p_{ab}|ab\rangle,$$

the reduced (or marginal) state of X alone is described by the probability vector

$$\sum_{(a,b)\in\Sigma\times\Gamma} p_{ab}|a\rangle,$$

For quantum state vectors, there is no analog — for a quantum state vector

$$|\phi\rangle \;=\; \sum_{(a,b)\in\Sigma\times\Gamma} \alpha_{ab}|ab\rangle,$$

the vector

$$|\phi\rangle \;=\; \sum_{(a,b)\in\Sigma\times\Gamma} \alpha_{ab}|a\rangle$$

is **not a quantum state vector** in general, and does not properly represent the concept of a **reduced or marginal state**. It could be, in fact, that this vector is the **zero vector**.

So, what we must do instead is turn to the **general description** of quantum information.

`The general description of quantum information provides with a meaningful way to define reduced quantum states that is analogous to the probabilistic setting.`

9.2.5 Unitary Operations :

In previous topics, we used the Cartesian product to treat individual systems as a larger, single system. Following the same line of thought, we can represent operations on multiple systems as unitary matrices acting on the state vector of this larger system.

In principle, any unitary matrix whose rows and columns correspond to the classical states of whatever system we're thinking about represents a valid quantum operation — and this holds true for compound systems whose classical state sets happen to be Cartesian products of the classical state sets of the individual systems.

Focusing on two systems, if X is a system having classical state set Σ and Y is a system having classical state set Γ, then the classical state set of the joint system (X,Y) is $\Sigma\times\Gamma$ — and therefore the set of operations that can be performed on this joint system are represented by **unitary matrices** whose rows and columns are placed in correspondence with the set $\Sigma \times \Gamma$.

The ordering of the rows and columns of these matrices is the same as the ordering used for quantum state vectors of the system (X,Y).

Example 1 : Let's consider a simple example involving two quantum systems, X and Y, where X has classical states $\Sigma = \{0, 1\}$ and Y has classical states $\Gamma = \{a, b\}$. The joint system (X, Y) has a classical state set given by the Cartesian product $\Sigma \times \Gamma = \{(0, a), (0, b), (1, a), (1, b)\}$.

Now, let's discuss how unitary matrices can represent operations on this joint system. In quantum mechanics, operations are often represented by unitary matrices, and these matrices act on the quantum state vector of the system.

For the joint system (X, Y), the quantum state vector can be expressed as:

$$|\psi\rangle = \alpha_{0a}|0a\rangle + \alpha_{0b}|0b\rangle + \alpha_{1a}|1a\rangle + \alpha_{1b}|1b\rangle$$

Here, $|\psi\rangle$ is a state vector representing the quantum state of the joint system. The coefficients α_{ij} are complex numbers associated with the classical states (i, j) in $\Sigma \times \Gamma$.

Now, let's consider a unitary matrix U that represents an operation on this joint system. The matrix U would have rows and columns corresponding to the classical states in $\Sigma \times \Gamma$. The ordering of rows and columns follows the same ordering used for quantum state vectors.

For simplicity, let's consider a specific unitary matrix U for a particular operation:

$$U = \begin{bmatrix} u_{00} & u_{01} & u_{02} & u_{03} \\ u_{10} & u_{11} & u_{12} & u_{13} \\ u_{20} & u_{21} & u_{22} & u_{23} \\ u_{30} & u_{31} & u_{32} & u_{33} \end{bmatrix}$$

In this matrix, each entry u_{ij} represents the amplitude for the transition from the state (i, j) to the state (k, l). For example, u_{01} represents the amplitude for the transition from state $(0, a)$ to $(1, a)$.

When this unitary matrix U acts on the state vector $|\psi\rangle$, it transforms the quantum state of the joint system according to the rules of quantum mechanics. The new state vector $|\psi^*\rangle$ is given by:

$$|\psi^*\rangle = U|\psi\rangle$$

This illustrates how unitary matrices can be used to represent operations on compound systems, and it provides a way to mathematically describe the evolution of the joint quantum state.

Example 2 : let us suppose that $\Sigma = \{1, 2, 3\}$ and $\Gamma = \{0, 1\}$, and recall that the standard convention for ordering the elements of the Cartesian product $\{1, 2, 3\} \times \{0, 1\}$ is $(1, 0)$, $(1, 1)$, $(2, 0)$, $(2, 1)$, $(3, 0)$, $(3, 1)$.

Here is an example of a unitary matrix representing an operation on (X, Y):

$$U = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{i}{2} & \frac{-1}{2} & 0 & 0 & \frac{-i}{2} \\ \frac{1}{2} & \frac{-1}{2} & \frac{1}{2} & 0 & 0 & \frac{-1}{2} \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & \frac{-i}{2} & \frac{-1}{2} & 0 & 0 & \frac{i}{2} \\ 0 & 0 & 0 & \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{pmatrix}$$

This unitary operation isn't important, it's just an example. To check that U is unitary, it suffices to compute:

$$U^\dagger U = \mathbb{I}$$

where $\mathbb{I}$ is Identity matrix of same order as of U.

The action of U on the standard basis vector $|\ 11\rangle$, for instance, is

$$U|11\rangle = \frac{1}{2}|10\rangle + \frac{i}{2}|11\rangle - \frac{1}{2}|20\rangle - \frac{i}{2}|30\rangle,$$

which we can see by examining the second column of U, considering our ordering of the set $\{1, 2, 3\} \times \{0, 1\}$.

The standard basis vectors for the Cartesian product set $\{1, 2, 3\} \times \{0, 1\}$ are formed by taking all possible combinations of elements from the individual sets. In this case, the standard basis vectors are the vectors with a single non-zero entry corresponding to one of the possible ordered pairs.

The ordered pairs in $\{1, 2, 3\} \times \{0, 1\}$ are: $(1, 0), (1, 1), (2, 0), (2, 1), (3, 0), (3, 1)$.

The standard basis vectors are then:

1. **Standard basis vector for** $(1, 0)$**:**

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

2. **Standard basis vector for** $(1, 1)$**:**

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

3. **Standard basis vector for** $(2, 0)$**:**

$$|2, 0\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

4. **Standard basis vector for** $(2, 1)$**:**

$$|2, 1\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

5. **Standard basis vector for** $(3, 0)$**:**

$$|3, 0\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

6. **Standard basis vector for** $(3,1)$**:**

$$|3,1\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

These vectors form the standard basis for the six-dimensional vector space spanned by the Cartesian product $\{1,2,3\}\times\{0,1\}$. Each vector corresponds to a unique ordered pair in the Cartesian product.

$$\implies U|11\rangle = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{i}{2} & \frac{-1}{2} & 0 & 0 & \frac{-i}{2} \\ \frac{1}{2} & \frac{-1}{2} & \frac{1}{2} & 0 & 0 & \frac{-1}{2} \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{2} & \frac{-i}{2} & \frac{-1}{2} & 0 & 0 & \frac{i}{2} \\ 0 & 0 & 0 & \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{pmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

After performing the matrix multiplication, we get:

$$U|11\rangle = \begin{bmatrix} \frac{1}{2} \\ \frac{i}{2} \\ \frac{-1}{2} \\ 0 \\ \frac{-i}{2} \\ 0 \end{bmatrix}$$

$$\implies U|11\rangle = \frac{1}{2}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \frac{i}{2}\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \frac{-1}{2}\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + 0\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \frac{-i}{2}\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} + 0\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\implies U|11\rangle = \frac{1}{2}|10\rangle + \frac{i}{2}|11\rangle + \frac{-1}{2}|20\rangle + 0|21\rangle + \frac{-i}{2}|30\rangle + 0|31\rangle$$

$$\implies U|11\rangle = \frac{1}{2}|10\rangle + \frac{i}{2}|11\rangle - \frac{1}{2}|20\rangle - \frac{i}{2}|30\rangle,$$

As with any matrix, it's possible to express U using the Dirac notation using 20 terms for the 20 nonzero entries of U.

However, if we did write down all of these terms rather than writing a 6×6 matrix, we might miss certain patterns that are evident from the matrix expression. Simply put, the Dirac notation is **not always the best choice** for how to represent matrices.

Important Note : Unitary operations on three or more systems work in a similar way, with the unitary matrices having rows and columns corresponding to the **Cartesian product of the classical state sets** of the systems.

We have already seen an example in this lesson: the three-qubit operation

$$\sum_{k=0}^{7} |(k+1) \ mod \ 8\rangle\langle k|$$

from before, where $|\ j\rangle$ means the three bit binary encoding of the number j, is unitary.

So, the expanded form of the given expression is:

$$|1\rangle\langle 0| + |2\rangle\langle 1| + |3\rangle\langle 2| + |4\rangle\langle 3| + |5\rangle\langle 4| + |6\rangle\langle 5| + |7\rangle\langle 6| + |0\rangle\langle 7|$$

Operations that are both unitary and represent deterministic operations are called reversible operations.

The conjugate transpose of this matrix can be written like this:

$$\sum_{k=0}^{7} |k\rangle\langle (k+1) \ mod \ 8| = \sum_{k=0}^{7} |(k-1) \ mod \ 8\rangle\langle k|.$$

The conjugate transpose would be:

$$\begin{aligned} &|0\rangle\langle 1| + |1\rangle\langle 2| + |2\rangle\langle 3| + |3\rangle\langle 4| + |4\rangle\langle 5| + |5\rangle\langle 6| + |6\rangle\langle 7| + |7\rangle\langle 0| \\ = |(-1 \ mod \ 8 =)7\rangle\langle 0| &+ |0\rangle\langle 1| + |1\rangle\langle 2| + |2\rangle\langle 3| + |3\rangle\langle 4| + |4\rangle\langle 5| + |5\rangle\langle 6| + |6\rangle\langle 7| \end{aligned}$$

This matrix represents the *reverse*, or in mathematical terms the **inverse**, of the original operation — which is what we expect from the conjugate transpose of a unitary matrix.

Unitary operations performed independently on individual systems : When unitary operations are performed independently on a collection of individual systems, the combined action of these independent operations is described by the tensor product of the unitary matrices that represent them.

That is, if $\mathsf{X}_1, \ldots, \mathsf{X}_n$ are quantum systems,$U_1, \ldots, U_n$ are unitary matrices representing operations on these systems, and the operations are performed independently on the systems, the combined action on $(\mathsf{X}_1, \ldots, \mathsf{X}_n)$ is represented by the matrix $U_1 \otimes \ldots \otimes U_n$. Once again, we find that the probabilistic and quantum settings are analogous in this regard.

`Tensor product of any collection of unitary matrices is unitary.`

Notice first that the conjugate transpose operation satisfies

$$(\mathsf{M}_1 \otimes \ldots \otimes \mathsf{M}_n)^\dagger \ = \mathsf{M}_1^\dagger \otimes \ldots \otimes \mathsf{M}_n^\dagger$$

for any collection of matrices $\mathsf{M}_1, \ldots, \mathsf{M}_n$.

- This can be checked by going back to the definition of the tensor product and of the conjugate transpose, and checking that each entry of the two sides of the equation are in agreement.

- This means that

$$(U_1 \otimes ... \otimes U_n)^\dagger (U_1 \otimes ... \otimes U_n) \; = (U_1^\dagger \otimes ... \otimes U_n^\dagger)(U_1 \otimes ... \otimes U_n)$$

- Because the tensor product of matrices is multiplicative, we find that

$$(U_1 \otimes ... \otimes U_n)^\dagger (U_1 \otimes ... \otimes U_n) \; = (U_1^\dagger U_1) \otimes ... \otimes (U_n^\dagger U_n) \; = \mathbb{I}_1 \otimes ... \otimes \mathbb{I}_n.$$

Here we have written $\mathbb{I}_1, ..., \mathbb{I}_n$ to refer to the matrices representing the identity operation on the systems $\mathsf{X}_1, ..., \mathsf{X}_n$ — which is to say that these are identity matrices whose sizes agree with the number of classical states of $\mathsf{X}_1, ..., \mathsf{X}_n$.

Finally, the tensor product $\mathbb{I}_1 \otimes ... \otimes \mathbb{I}_n$ is equal to the identity matrix, where we have a number of rows and columns that agrees with the product of the number of rows and columns of the matrices $\mathbb{I}_1, ..., \mathbb{I}_n$. We may view this larger identity matrix as representing the identity operation on the joint system $(\mathsf{X}_1, ..., \mathsf{X}_n)$.

In summary, we have the following sequence of equalities:

$$\begin{aligned}
(U_1 \otimes ... \otimes U_n)^\dagger (U_1 \otimes ... \otimes U_n) \; &= (U_1^\dagger \otimes ... \otimes U_n^\dagger)(U_1 \otimes ... \otimes U_n) \\
&= (U_1^\dagger U_1) \otimes ... \otimes (U_n^\dagger U_n) \\
&= \mathbb{I}_1 \otimes ... \otimes \mathbb{I}_n \\
&= \mathbb{I}
\end{aligned}$$

We therefore conclude that $U_1 \otimes ... \otimes U_n$ is **unitary**.

What if Unitary operation is applied only for one System ? An important situation that often arises is one in which a unitary operation is applied to just one system — or a proper subset of systems — within a larger joint system. For instance, suppose that X and Y are systems that we can view together as forming a single, compound system (X,Y), and we perform an operation just on the system X.

To be precise, let us suppose that U is a **unitary matrix** representing an operation on X, so that its **rows and columns have been placed in correspondence with the classical states of** X.

To say that we perform the operation represented by U just on the system X implies that we do nothing to Y, meaning that we independently perform U on X and the **identity operation** on Y.

That is, **"doing nothing"** to Y is equivalent to performing the identity operation on Y, which is represented by the identity matrix $\mathbb{I}_\mathsf{Y}$. (Here, by the way, the subscript Y tells us that $\mathbb{I}_\mathsf{Y}$ refers to the identity matrix having a number of rows and columns in agreement with the classical state set of Y.)

The operation on (X,Y) that is obtained when we perform U on X and do nothing to Y is therefore represented by the unitary matrix

$$U \otimes \mathbb{I}_\mathsf{Y}$$

For example, if X and Y are qubits, performing a **Hadamard operation** on Y (and doing nothing to Y) is equivalent to performing the operation

$$H \otimes \mathbb{I}_{\mathsf{Y}} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{-1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

on the joint system (X, Y).

Along similar lines, we may consider that an operation represented by a unitary matrix U is applied to Y and nothing is done to X, in which case the resulting operation on (X,Y) is represented by the unitary matrix

$$\mathbb{I}_{\mathsf{X}} \otimes U$$

For example, if we again consider the situation in which both X and Y are qubits and U is a **Hadamard operation**, the resulting operation on (X,Y) is represented by the matrix

$$\mathbb{I}_{\mathsf{X}} \otimes U = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

Not every unitary operation on a collection of systems $\mathsf{X}_1, \ldots, \mathsf{X}_n$ can be written as a **tensor product of unitary operations** $U_1 \otimes \ldots \otimes U_n$, just as **not every quantum state vector** of these systems is a **product state**.

For example, **neither the swap operation nor the controlled-NOT operation** on two qubits, which are described below, can be expressed as a **tensor product of unitary operations**.

SWAP Operations : Suppose that X and X are systems that share the same classical state set Σ. The swap operation on the pair (X,Y) is the operation that exchanges the contents of the two systems, but otherwise leaves the systems alone (*so that* X *remains on the left and* X *remains on the right*).

We will denote this operation as SWAP. It operates like this for every choice of classical states $a, b \in \Sigma$:

$$\boxed{\mathsf{SWAP}|a\rangle|b\rangle = |b\rangle|a\rangle}$$

One way to write the matrix associated with this operation using the **Dirac notation** is as follows:

$$\boxed{\mathsf{SWAP} = \sum_{c,d \in \Sigma} |c\rangle\langle d| \otimes |d\rangle\langle c|}$$

It may not be immediately clear that this matrix represents SWAP, but we can check it satisfies the condition $\mathsf{SWAP}|a\rangle|b\rangle = |b\rangle|a\rangle$ for every choice of classical states $a, b \in \Sigma$.

As a simple example, when X and Y are **qubits**, we find that

$$\boxed{\mathsf{SWAP} \;=\; \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}$$

Controlled Unitary Operations : Now let us suppose that Q is a **qubit** and R is an **arbitrary system**, having whatever classical state set we wish.

For every unitary operation U acting on the system R, a ***controlled*** U operation is a unitary operation on the pair (Q,R) defined as follows:

$$CU \;= |0\rangle\langle 0| \otimes \mathbb{I}_{\mathsf{R}} + |1\rangle\langle 1| \otimes U$$

For example, if R is also a **qubit** and we think about the **Pauli** X operation (**Bit-Flip Operation** or **CNOT gate**) on R, then a controlled-X operation is given by

$$CX = |0\rangle\langle 0| \otimes \mathbb{I}_{\mathsf{R}} + |1\rangle\langle 1| \otimes X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\begin{aligned} CX &= \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{aligned}$$

Where ***Pauli-X*** is

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

If instead we consider the **Pauli** Z operation (***Phase-flip Operation***) on R in place of the X operation, we obtain this operation:

$$CZ = |0\rangle\langle 0| \otimes \mathbb{I}_{\mathsf{R}} + |1\rangle\langle 1| \otimes Z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

Where ***Pauli-Z*** is

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

If instead we take R to be two qubits, and we take U to be the SWAP operation between these two qubits, we obtain this operation:

$$\text{CSWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

- This operation is also known as a **Fredkin operation** (or, more commonly, a Fredkin gate), named for Edward Fredkin.

Its action on standard basis states can be described as follows:

$$\text{CSWAP} \mid 0bc\rangle = \mid 0bc\rangle$$
$$\text{CSWAP} \mid 1bc\rangle = \mid 1cb\rangle$$

Finally, **controlled-controlled-NOT** operation, which we may denote as CCX, is called a **Toffoli operation** (or Toffoli gate), named for Tommaso Toffoli. Its matrix representation looks like this:

$$CCX = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

We may alternatively express it using the Dirac notation as follows:

$$CCX = \Big(\mid 00\rangle\langle 00 \mid + \mid 01\rangle\langle 01 \mid + \mid 10\rangle\langle 10 \mid \Big) \otimes \mathbb{I} + \mid 11\rangle\langle 11 \mid \otimes X.$$

9.2.6 Qiskit Examples :

In the previous topics, we learned about Qiskit's **Statevector** and **Operator** classes, and used them to simulate quantum systems. In this section, we'll use them to explore the behavior of multiple systems. We'll start by importing these classes, as well as the **square root** function from **NumPy** .

[27]:
```
from qiskit.quantum_info import Statevector, Operator
from numpy import sqrt
```

Tensor Products : The Statevector class has a tensor method which returns the tensor product of itself and another Statevector .

For example, below we create two state vectors representing $|0\rangle$ and $|1\rangle$, and use the tensor method to create a new vector, $|0\rangle \otimes |1\rangle$.

[28]:
```
zero, one = Statevector.from_label("0"), Statevector.from_label("1")
#tensor product of zero x one
zero.tensor(one).draw("latex")
```

[28]:

$$|01\rangle$$

[29]:
```
import numpy as np
print(np.array2string(np.real(zero.tensor(one).data).astype(int), separator=',␣
↪'))
```

```
[0, 1, 0, 0]
```

In another example below, we create state vectors representing the $|+\rangle$ and $\frac{1}{\sqrt{2}}(|0\rangle + i\,|1\rangle)$ states, and combine them to create a new state vector. We'll assign this new vector to the variable psi - (ψ).

[30]:
```
plus = Statevector.from_label("+")

plus.draw("latex")
```

[30]:

$$\frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}}{2}|1\rangle$$

[31]:
```
i_state = Statevector([1 / sqrt(2), 1j / sqrt(2)])

i_state.draw("latex")
```

[31]:

$$\frac{\sqrt{2}}{2}|0\rangle + \frac{\sqrt{2}i}{2}|1\rangle$$

[32]:
```
psi = plus.tensor(i_state)

psi.draw("latex")
```

[32]:

$$\frac{1}{2}|00\rangle + \frac{i}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{i}{2}|11\rangle$$

The Operator class also has a tensor method. In the example below, we create the X and I gates and display their tensor product.

[33]:
```
X = Operator([[0, 1], [1, 0]])
I = Operator([[1, 0], [0, 1]])

X.draw("text")
```

[33]:
```
[[0.+0.j,1.+0.j],
 [1.+0.j,0.+0.j]]
```

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

[34]:
```
I.draw('text')
```

[34]:
```
[[1.+0.j,0.+0.j],
 [0.+0.j,1.+0.j]]
```

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

[35]:
```
X.tensor(I)
```

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

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

We can then treat these compound states and operations as we did single systems in the previous lesson. For example, in the cell below we calculate

$$(I \otimes \mathsf{X}) \mid \psi\rangle$$

for the state **psi** we defined above.

(The ^ operator **tensors matrices together**.)

[36]:
```
psi.evolve(I ^ X).draw("latex")
```

[36]:

$$\frac{i}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{i}{2}|10\rangle + \frac{1}{2}|11\rangle$$

[37]:
```
print(I^X)
```

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

value of **I^X** is

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

Here we already have **Psi** as $\frac{1}{2}|00\rangle + \frac{i}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{i}{2}|11\rangle$

$$\implies |\psi\rangle = \frac{1}{2}|00\rangle + \frac{i}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{i}{2}|11\rangle = \frac{1}{2}\begin{bmatrix}1\\0\\0\\0\end{bmatrix} + \frac{i}{2}\begin{bmatrix}0\\1\\0\\0\end{bmatrix} + \frac{1}{2}\begin{bmatrix}0\\0\\1\\0\end{bmatrix} + \frac{i}{2}\begin{bmatrix}0\\0\\0\\1\end{bmatrix} = \begin{bmatrix}\frac{1}{2}\\\frac{i}{2}\\\frac{1}{2}\\\frac{i}{2}\end{bmatrix}$$

$$\mathrm{I\,\hat{}\,X} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\implies \text{Result} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix}\frac{1}{2}\\\frac{i}{2}\\\frac{1}{2}\\\frac{i}{2}\end{bmatrix}$$

Performing the matrix multiplication, you get:

$$\text{Result} = \begin{bmatrix} i/2 \\ 1/2 \\ i/2 \\ 1/2 \end{bmatrix}$$

$$\implies \boxed{(I \otimes \mathrm{X}) \mid \psi\rangle = \begin{bmatrix} i/2 \\ 1/2 \\ i/2 \\ 1/2 \end{bmatrix} = \frac{i}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{i}{2}|10\rangle + \frac{1}{2}|11\rangle}$$

Below, we create a CX operator and calculate $CX \mid \psi\rangle$.

[38]:
```
CX = Operator(
    [
        [1, 0, 0, 0],
```

```
        [0, 1, 0, 0],
        [0, 0, 0, 1],
        [0, 0, 1, 0],
    ]
)

psi.evolve(CX).draw("latex")
```

[38]:

$$\frac{1}{2}|00\rangle + \frac{i}{2}|01\rangle + \frac{i}{2}|10\rangle + \frac{1}{2}|11\rangle$$

Partial Measurements : In the examples of single systems, we used the ***measure*** method to simulate a measurement of the quantum state vector.

- This method returns two items: the **simulated measurement result**, and the **new Statevector** given this measurement.

By default, ***measure*** measures all qubits in the state vector, but we can provide a list of integers to only measure the qubits at those indices. To demonstrate, the cell below creates the state

$$W = \frac{1}{\sqrt{3}}(|001\rangle + |010\rangle + |100\rangle)$$

(Note that Qiskit is primarily designed for use with qubit-based quantum computers. As such, Statevector will try to interpret any vector with 2^n elements as a system of n qubits. we can override this by passing a **dims** argument to the constructor. For example, **dims=(4,2)** would tell Qiskit the system has one **four-level system**, and one **two-level system** (qubit).)

[39]:
```
W = Statevector([0, 1, 1, 0, 1, 0, 0, 0] / sqrt(3))
W.draw("latex")
```

[39]:

$$\frac{\sqrt{3}}{3}|001\rangle + \frac{\sqrt{3}}{3}|010\rangle + \frac{\sqrt{3}}{3}|100\rangle$$

The cell below simulates a measurement on the rightmost qubit (which has index 0). The other two qubits are not measured.

[40]:
```
result, new_sv = W.measure([0])  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 0
State after measurement:
```

[40]:

$$\frac{\sqrt{2}}{2}|010\rangle + \frac{\sqrt{2}}{2}|100\rangle$$

Try running the cell a few times to see different results. Notice that measuring a **1** means that we know both the other qubits are $|0\rangle$, but measuring a **0** means the remaining two qubits are in the state $\frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$.

[41]:
```
result, new_sv = W.measure([0])  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 0
State after measurement:
```

[41]:

$$\frac{\sqrt{2}}{2}|010\rangle + \frac{\sqrt{2}}{2}|100\rangle$$

Here $|abc\rangle$ means a -> left most qubit, b -> middle qubit and c -> right most qubit.

This means here we are measuring **qubit[0]**,$(|q_2 q_1 q_0\rangle)$ and in total we only have 3 states.

Here if the $q_0 = 1$ it obviously mean $|001\rangle$ as it is the only state having **1** at q_0.

[42]:
```
result, new_sv = W.measure([1])  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 0
State after measurement:
```

[42]:

$$\frac{\sqrt{2}}{2}|001\rangle + \frac{\sqrt{2}}{2}|100\rangle$$

This measurement means we are measuring **q_1** to be in state **1**. That is only one state is having $q_1 = 1$ so the superposition collapses to the corresponding state.

But if we want $q_1 = 0$ then there are two states which have $q_1 = 0$ in common, so the final state collapses to the position in between those two states and this position is determined by the multiplying factors each state has.

[43]:
```
result, new_sv = W.measure([1])  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 0
State after measurement:
```

[43]:

$$\frac{\sqrt{2}}{2}|001\rangle + \frac{\sqrt{2}}{2}|100\rangle$$

[44]:
```
result, new_sv = W.measure([2])  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
```

```
new_sv.draw("latex")
```

```
Measured: 0
State after measurement:
```

[44]:

$$\frac{\sqrt{2}}{2}|001\rangle + \frac{\sqrt{2}}{2}|010\rangle$$

[45]:
```
result, new_sv = W.measure([2])  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 0
State after measurement:
```

[45]:

$$\frac{\sqrt{2}}{2}|001\rangle + \frac{\sqrt{2}}{2}|010\rangle$$

[46]:
```
result, new_sv = W.measure()  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 100
State after measurement:
```

[46]:

$$|100\rangle$$

[47]:
```
result, new_sv = W.measure()  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 001
State after measurement:
```

[47]:

$$|001\rangle$$

[48]:
```
result, new_sv = W.measure()  # measure qubit 0
print(f"Measured: {result}\nState after measurement:")
new_sv.draw("latex")
```

```
Measured: 100
State after measurement:
```

[48]:

$$|100\rangle$$

10 Quantum Circuits :

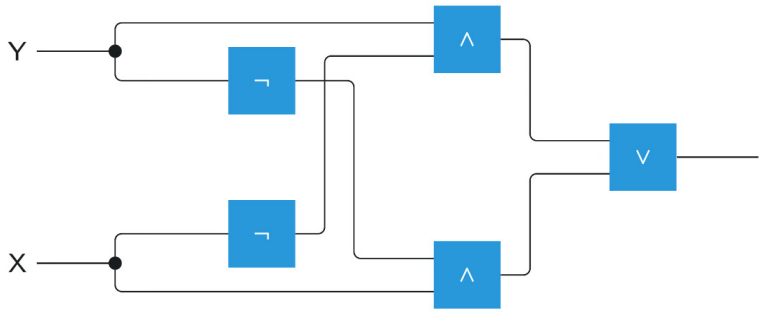
10.1 Circuits :

In computer science, *circuits* are models of computation in which information is carried by wires through a network of gates, which represent operations that transform the information carried by the wires. Quantum circuits are just one example of a model of computation based on this more general concept.

Although the word **"circuit"** often refers to a circular path, circular paths aren't actually allowed in the most common circuit models of computation. That is to say, we usually study **acyclic circuits** when we're thinking about circuits as computational models. Quantum circuits follow this pattern; although we could run a quantum circuit as many times as we like, a quantum circuit itself represents a finite sequence of operations that cannot contain feedback loops.

10.1.1 Boolean Circuits :

Here is an example of a (classical) Boolean circuit, where the wires carry binary values and the gates represent Boolean logic operations:



The flow of information along the wires goes from left to right:

- the wires on the **left-hand side** of the figure labeled X and Y are *input* bits, which can each be set to whatever binary value we choose,
- and the wire on the **right-hand side** is the *output*.
- The intermediate wires take whatever values are **determined by the gates**, which are evaluated from left to right.

The gates are ***AND*** gates (labeled $\wedge$), ***OR*** gates (labeled $\vee$), and ***NOT*** gates (labeled $\neg$). The functions computed by these gates will likely be familiar to many readers, but here they are represented by tables of values:

a	$\neg a$
0	1
1	0

ab	$a \wedge b$
00	0
01	0
10	0
11	1

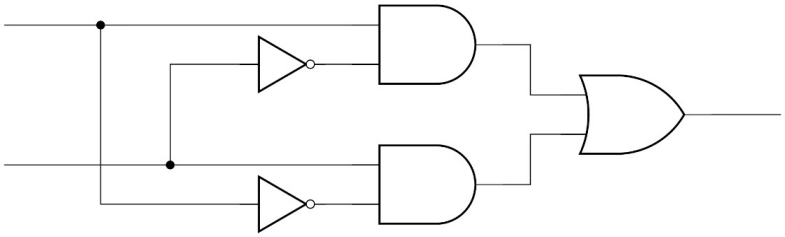
ab	$a \vee b$
00	0
01	1
10	1
11	1

The two small circles on the wires just to the right of the names X and Y represent **fanout**

operations, which simply create a copy of whatever value is carried on the wire on which they appear, so that this value can be input into multiple gates.

Fanout operations are not always considered to be gates in the classical setting — sometimes they are treated as if they are "free" in some sense — but when we discuss how ordinary Boolean circuits can be converted into equivalent quantum circuits, ***we must classify fanout operations explicitly as gates*** and account for them correctly.

Here is the same circuit illustrated in a style more common in electrical engineering, which uses conventional symbols for the AND, OR, and NOT gates:

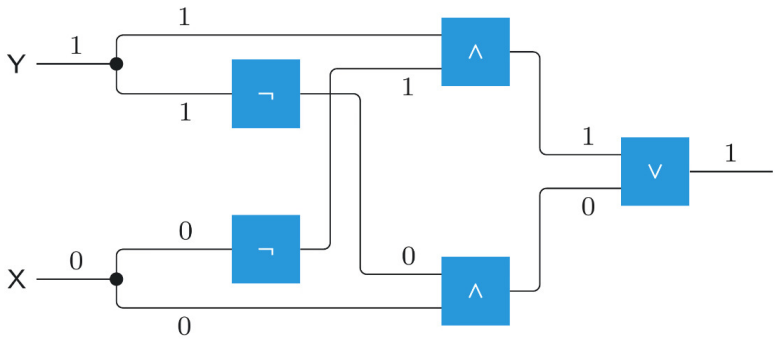


We won't use this style or these particular gate symbols further, but we do use different symbols to represent gates in quantum circuits, which we'll explain as we encounter them.

The particular circuit in this example computes the exclusive-OR (or XOR for short), which is denoted by the symbol $\oplus$:

ab	$a \oplus b$
00	0
01	1
10	1
11	0

In the next diagram we consider just one choice for the inputs: $\mathsf{X} = 0$ and $\mathsf{Y} = 1$. Each wire is labeled by value it carries so you can follow the operations. The output value is 1 in this case, which is the correct value for the XOR : $0 \oplus 1 = 1$.

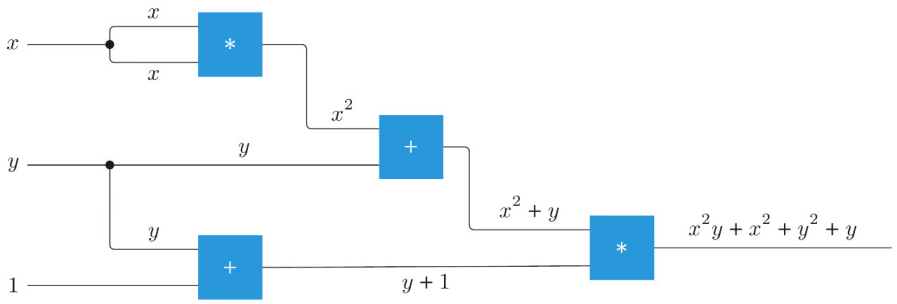


Other types of circuits : As suggested above, the notion of a circuit in computer science is very general. For example, circuits whose wires carry values other than 0 and 1 are sometimes studied, as are gates representing different choices of operations.

In arithmetic circuits, for instance, the wires may carry integer values and the gates may represent arithmetic operations, such as **addition and multiplication**. The following figure depicts an

arithmetic circuit that takes two variable input values (x and y) as well as a third input set to the value 1.

The values carried by the wires, as functions of the values x and y, are shown in the figure.



We can also consider circuits that incorporate randomness, such as ones where gates represent probabilistic operations.

10.1.2 Quantum Circuits :

In the quantum circuit model, wires represent qubits and gates represent operations acting on these qubits. We'll focus for now on operations we've encountered so far, namely unitary operations and standard basis measurements. As we learn about other sorts of quantum operations and measurements, we'll enhance our model accordingly.

Here's a simple example of a quantum circuit:

The order of applying operation on the qubit is from **left to right**. Inintially, *Hadamard-gate* (**H**) is performed on the qubit, the resulting state is again performed with *Phase Operation-gate* (**S**) and so on...

Corresponding Operators are :

$$\mathsf{H} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \implies HadamardGate$$

$$\mathsf{S} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \implies Phase - Operator$$

$$\mathsf{T} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} \end{pmatrix} \implies ToffoliGate$$

When all the mentioned operators are performed on qubit in the same order, then the resultant

operator is given as follows.

$$\mathsf{THSH} = \begin{pmatrix} \frac{1+i}{2} & \frac{1-i}{2} \\ \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \end{pmatrix}$$

This means the action on the qubit X is determined by this unitary matrix.

Sometimes we wish to explicitly indicate the input or output states to a circuit. For example, if we apply the operation THSH to the state $|0\rangle$, we obtain the state $\frac{1+i}{2}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. We may indicate this as follows:

Quantum circuits often have all qubits initialized to $|0\rangle$, as we have in this case, but there are also cases where we wish to set the input qubits to different states.

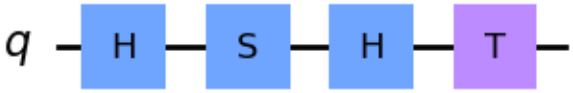
Now let's see how we can specify this circuit in Qiskit, beginning with the imports needed for the current section.

[49]:
```
from qiskit import QuantumCircuit, QuantumRegister, ClassicalRegister
from qiskit.primitives import Sampler
from qiskit.visualization import plot_histogram
```

To begin, we can create the circuit as follows, by sequentially adding gates from left to right.

[50]:
```
circuit = QuantumCircuit(1) #Quantum Circuit of 1 Qubit

circuit.h(0) #hadamard gate
circuit.s(0) #Phase - Operator
circuit.h(0) #Hadamard -gate
circuit.t(0) #toffoli-gate

display(circuit.draw("mpl")) #printing the circuit
```

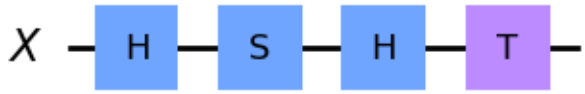
```
/home/cdac/anaconda3/envs/qiskit/lib/python3.10/site-
packages/qiskit/visualization/circuit/matplotlib.py:266: FutureWarning: The
default matplotlib drawer scheme will be changed to "iqp" in a following
release. To silence this warning, specify the current default explicitly as
style="clifford", or the new default as style="iqp".
  self._style, def_font_ratio = load_style(self._style)
```



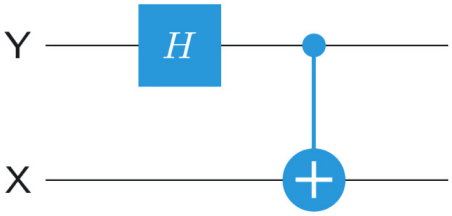
The default names for qubits in Qiskit are q_0, q_1, q_2, etc., and when there is just a single qubit like in our example, the default name is q rather than q_0.

- If we wish to choose our own name we can do this using the QuantumRegister class like this:

[51]:
```
X = QuantumRegister(1, "X") #1st Parameter for number of Qubits, 2nd for␣
 ↪assigning a name to the Qubits
circuit = QuantumCircuit(X)

circuit.h(X)
circuit.s(X)
circuit.h(X)
circuit.t(X)

display(circuit.draw("mpl")) #mpl is used to print the circuit with help of␣
 ↪matplotlib (can be ignored if needed)
```



Here is another example of a quantum circuit, this time with two qubits:



As always, the gate labeled H refers to a **Hadamard operation**, while the second gate is a **two-qubit** gate: it's the **controlled-NOT** operation, where the **solid circle represents the control qubit** and the **circle resembling the symbol $\oplus$ denotes the target qubit**.

Before examining this circuit in greater detail and explaining what it does, it is imperative that we clarify how qubits are ordered in quantum circuits, which connects with the convention that Qiskit uses for naming and ordering systems mentioned briefly in the previous lesson.

Qiskit's qubit ordering convention for circuits : In Qiskit, the top-most qubit in a circuit diagram has index 0 and corresponds to the right-most position in a tuple of qubits (or in a string,

Cartesian product, or tensor product corresponding to this tuple).

The second-from-top qubit has index 1, and corresponds to the position second-from-right in a tuple, and so on, down to the bottommost qubit, which has the highest index, and corresponds to the leftmost position in a tuple.

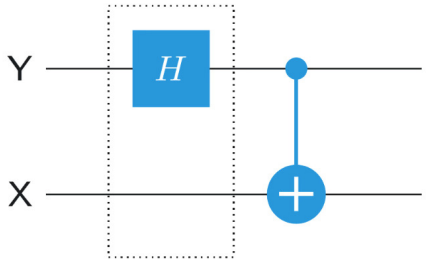
In particular, Qiskit's default names for the qubits in an n-qubit circuit are represented by the n-tuple $(q_{n-1}, \ldots, q_0)$, with q_0 being the qubit on the top and $q_n - 1$ on the bottom in quantum circuit diagrams.

Further information on this ordering convention can be found on the Bit-ordering in Qiskit documentation page. https://docs.quantum-computing.ibm.com/build/bit-ordering

Although we will often deviate from the specific default names $q_{n-1}, \ldots, q_0$ used for qubits by Qiskit, we will follow this ordering convention throughout this course for interpreting circuit diagrams.

Thus, our interpretation of the circuit above is that it describes an operation on a pair of qubits (X, Y) — and if the input to the circuit is a quantum state $|\,\psi\rangle\,|\,\phi\rangle$, then this means that the lower qubit X starts in the state $|\,\psi\rangle$ and the upper qubit Y starts in the state $|\,\phi\rangle$. To understand what the circuit does, we can go from left to right through its operations.

1. **The first operation is a Hadamard operation on Y:**

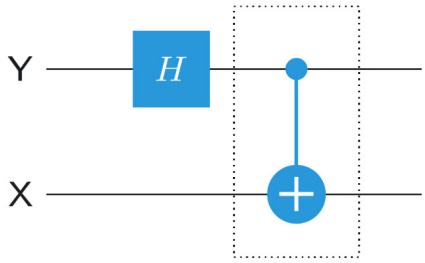


When applying a gate to a single qubit like this, nothing happens to the other qubits — and nothing happening is equivalent to the identity operation being performed. In our circuit there is just one other qubit, X, so the dotted rectangle in the figure above represents this operation:

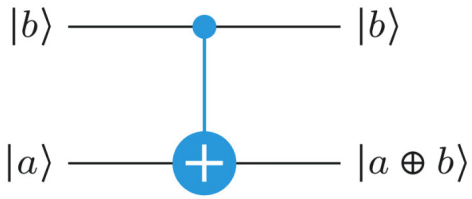
$$\mathbb{I} \otimes \mathsf{H} = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

Note that the identity matrix is on the left of the tensor product and H is on the right, which is consistent with Qiskit's ordering convention.

2. **The second operation is the controlled-NOT operation, where Y is the control and X is the target:**



The controlled-NOT gate's action on standard basis states is as follows:



Given that we order the qubits as (X, Y), the matrix representation of the controlled-NOT gate is this:

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

The unitary operation of the entire circuit, which we'll call U, is the composition of the operations:

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

In particular, recalling our notation for Bell states,

$$\begin{aligned} |\phi^+\rangle &= \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle \\ |\phi^-\rangle &= \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|11\rangle \\ |\psi^+\rangle &= \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle \\ |\psi^-\rangle &= \frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|10\rangle \end{aligned}$$

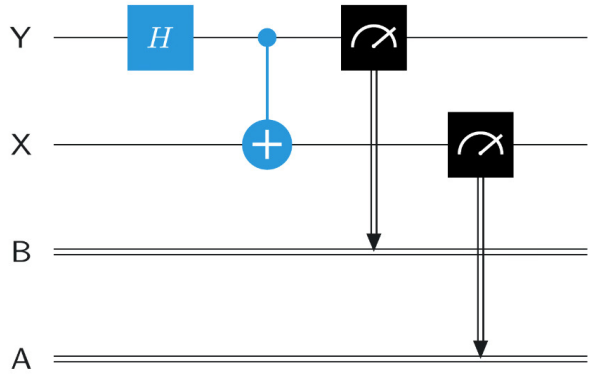
we get that

$$\begin{aligned} U|00\rangle &= |\phi^+\rangle \\ U|01\rangle &= |\phi^-\rangle \\ U|10\rangle &= |\psi^+\rangle \\ U|11\rangle &= -|\psi^-\rangle \end{aligned}$$

So, this circuit gives us a way to create the state $|\phi^+\rangle$ if we run it on two qubits initialized to $|\,00\rangle$.

- More generally, it gives us a way to convert the standard basis to the Bell basis.
- (The -1 phase factor on the last state, $-|\psi^-\rangle$, could be eliminated if we wanted, but this would require a change to the circuit. For instance, we could add a controlled-Z gate at the beginning or a swap gate at the end to eliminate the minus sign without affecting the circuit's action on the other three standard basis states.)

In general, **quantum circuits can contain any number of qubit wires**. We may also include classical bit wires, which are indicated by **double lines** like in this example:

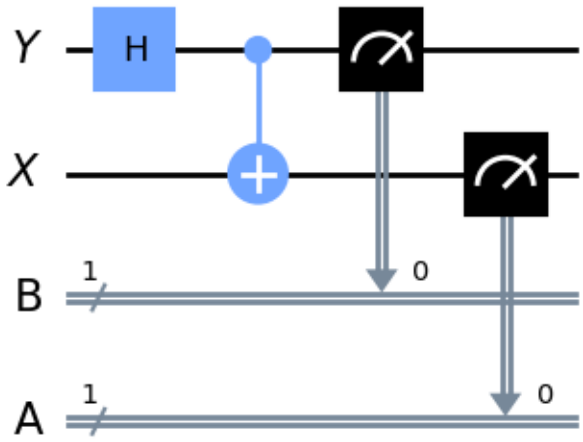


In this circuit we have a Hadamard gate and a controlled-NOT gate on two qubits X and Y, just like in the previous example.

We also have two classical bits, A and B, as well as two measurement gates.

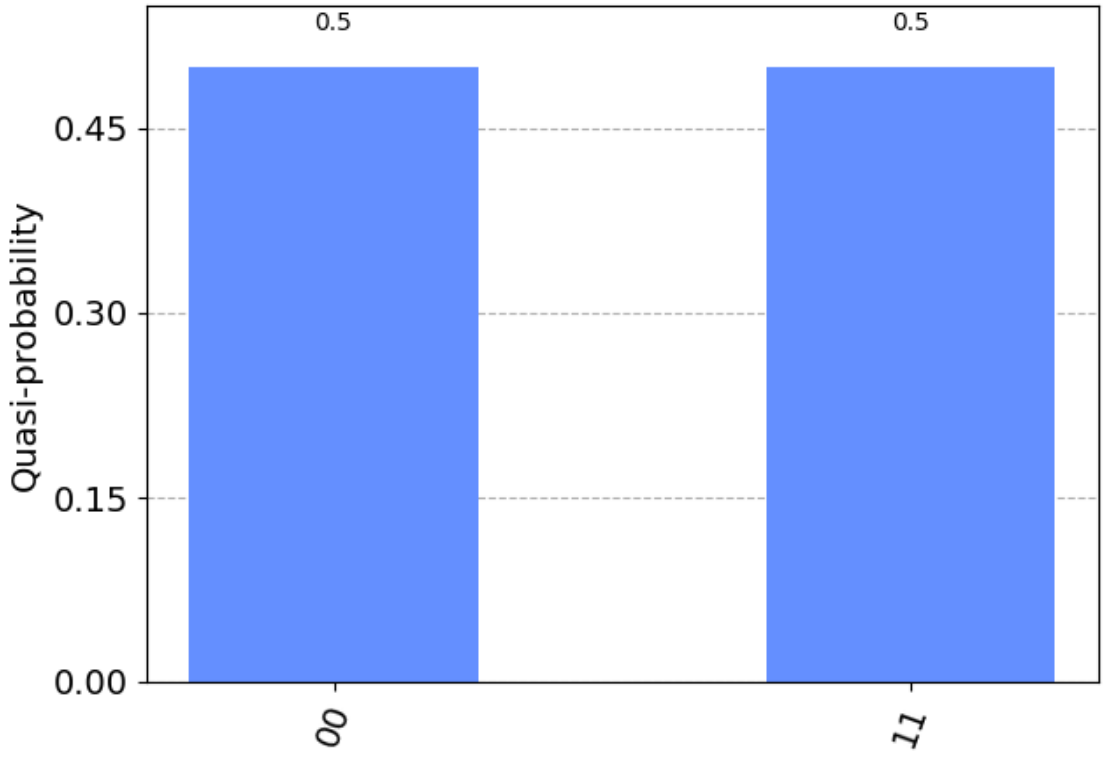
The measurement gates represent standard basis measurements: the qubits are changed into their post-measurement states, while the measurement outcomes are overwritten onto the classical bits to which the arrows point.

[52]:
```
X = QuantumRegister(1, "X")
Y = QuantumRegister(1, "Y")
A = ClassicalRegister(1, "A")
B = ClassicalRegister(1, "B")

circuit = QuantumCircuit(Y, X, B, A)
circuit.h(Y)
circuit.cx(Y, X)
circuit.measure(Y, B)
circuit.measure(X, A)

display(circuit.draw("mpl"))
```



The circuit can be simulated using the **`Sampler`** primitive.

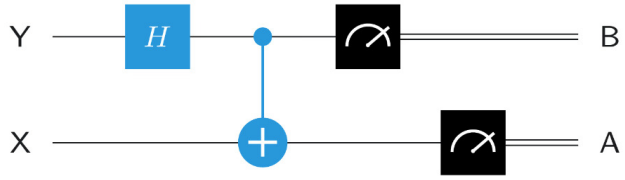
[53]:
```
results = Sampler().run(circuit).result()
statistics = results.quasi_dists[0].binary_probabilities()
display(plot_histogram(statistics))
```



Sometimes it is convenient to depict a measurement as a gate that takes a qubit as input and outputs a classical bit (as opposed to outputting the qubit in its post-measurement state and writing the result to a separate classical bit).

- This means the measured qubit has been discarded and can safely be ignored thereafter.

For example, the following circuit diagram represents the same process as the one in the previous diagram, but where we ignore X and Y after measuring them:

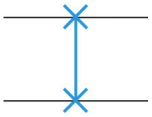


Here are some symbols for common gates:

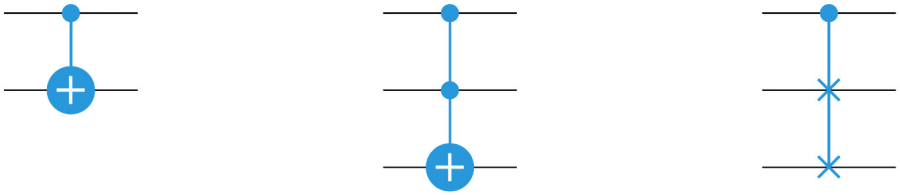
- Single-qubit gates are generally shown as squares with a letter indicating which operation it is,

like this:

- Not gates (also known as X gates) are also sometimes denoted by a circle around a plus sign:

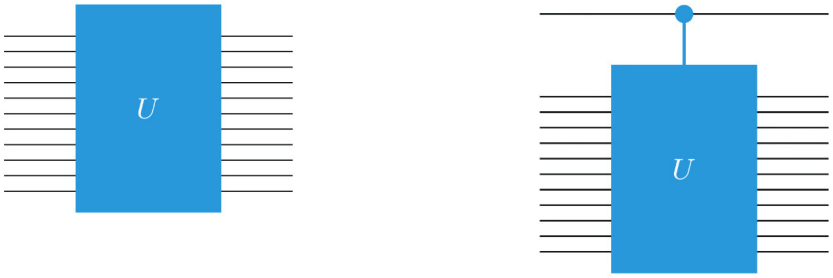


- Swap gates are denoted as follows:
- Controlled-gates, meaning gates that describe controlled-unitary operations, are denoted by a filled-in circle (indicating the control) connected by a vertical line to whatever operation is being controlled. For instance, **controlled-NOT** gates, **controlled-controlled-NOT (or Toffoli)** gates, and **controlled-swap (Fredkin)** gates are denoted like this:



- Arbitrary unitary operations on multiple qubits may be viewed as gates. They are depicted by rectangles labeled by the name of the unitary operation. For instance, here is a depiction

of an (unspecified) unitary operation U as a gate, along with a controlled version of this gate:



10.1.3 Inner products, orthonormality, and projections :

To better prepare ourselves to explore the capabilities and limitations of quantum circuits, we now introduce some additional mathematical concepts — namely the **inner product** between vectors (and its connection to the Euclidean norm), the notions of **orthogonality and orthonormality** for sets of **vectors**, and **projection matrices**, which will allow us to introduce a handy generalization of standard basis measurements.

Inner products: When we use the Dirac notation to refer to an arbitrary column vector as a ket, such as

$$|\,\psi\rangle = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix},$$

the corresponding bra vector is the ***conjugate transpose*** of this vector:

$$\langle\psi| = (|\psi\rangle)^\dagger = (\overline{\alpha_1} \quad \overline{\alpha_2} \quad \ldots \quad \overline{\alpha_n}). \tag{1}$$

Alternatively, if we have some classical state set Σ in mind, and we express a column vector as a ket, such as

$$|\psi\rangle = \sum_{a\in\Sigma} \alpha_a |a\rangle,$$

then the corresponding row (or bra) vector is the conjugate transpose

$$\langle\psi| = \sum_{a\in\Sigma} \overline{\alpha_a} \langle a| \tag{2}$$

We also observed that the product of a bra vector and a ket vector, viewed as matrices either having a single row or a single column, results in a scalar. Specifically, if we have two (column) vectors

$$|\psi\rangle = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix} \quad \text{and} \quad |\phi\rangle = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix}$$

so that the row vector $\langle \psi \mid$ is as in equation (1), then

$$\langle \psi | \phi \rangle = \langle \psi || \phi \rangle = \begin{pmatrix} \overline{\alpha_1} & \overline{\alpha_2} & \ldots & \overline{\alpha_n} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix} = \overline{\alpha_1}\beta_1 + \ldots + \overline{\alpha_n}\beta_n.$$

Alternatively, if we have two column vectors that we have written as

$$|\psi\rangle = \sum_{a \in \Sigma} \alpha_a |a\rangle \quad \text{and} \quad |\phi\rangle = \sum_{b \in \Sigma} \beta_b |b\rangle,$$

so that $\langle \psi |$ is the row vector (2), we find that

$$\begin{aligned} \langle \psi | \phi \rangle &= \langle \psi || \phi \rangle \\ &= \Bigg(\sum_{a \in \Sigma} \overline{\alpha_a} \langle a | \Bigg) \Bigg(\sum_{b \in \Sigma} \beta_b |b\rangle \Bigg) \\ &= \sum_{a \in \Sigma} \sum_{b \in \Sigma} \overline{\alpha_a} \beta_b \langle a | b \rangle \\ &= \sum_{a \in \Sigma} \overline{\alpha}_a \beta_a, \end{aligned}$$

where the last equality follows from the observation that $\langle a \mid a \rangle = 1$ and $\langle a \mid b \rangle = 0$ for classical states a and b satisfying $a \neq b$.

The value $\langle \psi \mid \phi \rangle$ is called the ***inner product*** between the vectors $\mid \psi \rangle$ and $\mid \phi \rangle$.

- Inner products are critically important in quantum information and computation.

We would not get far in understanding quantum information at a mathematical level without this fundamental notion.

Let us now collect together some basic facts about inner products of vectors.

1. Relationship to the Euclidean norm :

The inner product of any vector

$$|\psi\rangle = \sum_{a \in \Sigma} \alpha_a |a\rangle$$

with itself is

$$\langle \psi \mid \psi \rangle = \sum_{a \in \Sigma} \overline{\alpha_a} \alpha_a = \sum_{a \in \Sigma} |\alpha_a|^2 = \Big\| |\psi\rangle \Big\|^2$$

Thus, the Euclidean norm of a vector may alternatively be expressed as

$$\Big\| |\psi\rangle \Big\| = \sqrt{\langle \psi \mid \psi \rangle}$$

Notice that the Euclidean norm of a vector must always be a **nonnegative real number**. Moreover, the only way the Euclidean norm of a vector can be equal to zero is if every one of the entries is equal to zero, which is to say that the vector is the zero vector.

We can summarize these observations like this: for every vector $|\,\psi\rangle$ we have

$$\langle\psi|\psi\rangle \geq 0,$$

with $\langle\psi|\psi\rangle = 0$ if and only if $|\psi\rangle = 0$. This property of inner product is sometimes referred to as ***positive definiteness***.

2. Conjugate Symmetry :

For any two vectors

$$|\psi\rangle = \sum_{a\in\Sigma} \alpha_a |a\rangle \quad \text{and} \quad |\phi\rangle = \sum_{b\in\Sigma} \beta_b |b\rangle$$

we have

$$\langle\psi|\phi\rangle = \sum_{a\in\Sigma} \overline{\alpha_a}\beta_a \quad \text{and} \quad \langle\phi|\psi\rangle = \sum_{a\in\Sigma} \overline{\beta_a}\alpha_a,$$

and therefore

$$\overline{\langle\psi|\phi\rangle} = \langle\phi|\psi\rangle.$$

3. Linearity in the second argument (and conjugate linearity in the first).

Let us suppose that $|\,\psi\rangle, |\phi_1\rangle, |\phi_2\rangle$ are vectors and α_1 and α_2 are complex numbers. If we define a new vector

$$|\phi\rangle = \alpha_1|\phi_1\rangle + \alpha_2|\phi_2\rangle,$$

then

$$\langle\psi|\phi\rangle = \langle\psi|(\alpha_1|\phi_1\rangle + \alpha_2|\phi_2\rangle) = \alpha_1\langle\psi|\phi_1\rangle + \alpha_2\langle\psi|\phi_2\rangle.$$

That is to say, the inner product is ***linear*** in the second argument. This can be verified either through the formulas above or simply by noting that matrix multiplication is linear in each argument (and specifically in the second argument).

Combining this fact with conjugate symmetry reveals that the inner product is ***conjugate linear*** in the first argument. That is, if $|\,\psi_1\rangle, |\,\psi_2\rangle, |\,\phi\rangle$ are vectors and α_1 and α_2 are complex numbers, and we define

$$|\psi\rangle = \alpha_1|\psi_1\rangle + \alpha_2|\psi_2\rangle,$$

then,

$$\langle\psi|\phi\rangle = (\overline{\alpha_1}\langle\psi_1| + \overline{\alpha_2}\langle\psi_2|)|\phi\rangle = \overline{\alpha_1}\langle\psi_1|\phi\rangle + \overline{\alpha_2}\langle\psi_2|\phi\rangle.$$

4. The Cauchy-Schwarz inequality.

For every choice of vectors $|\,\phi\rangle$ and $|\,\psi\rangle$ having the same number of entries, we have

$$|\langle\psi|\phi\rangle| \leq \Big\||\psi\rangle\Big\| \Big\||\phi\rangle\Big\|.$$

This is an incredibly handy inequality that gets used quite extensively in quantum information (and in many other fields of study).

Orthogonal and Orthonormal Sets : Two vectors $|\phi\rangle$ and $|\psi\rangle$ are said to be **orthogonal** if their inner product is zero:

$$\langle\psi|\phi\rangle = 0.$$

Geometrically, we can think about orthogonal vectors as vectors at right angles to each other.

A set of vectors $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$ is called an orthogonal set if every vector in the set is orthogonal to every other vector in the set. That is, this set is orthogonal if

$$\langle\psi_j|\psi_k\rangle = 0$$

for all choices of $j, k \in \{1, \ldots, m\}$ for which $j \neq k$.

A set of vectors $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$ is called an ***orthonormal*** set if it is an orthogonal set and, in addition, every vector in the set is a ***unit vector***. Alternatively, this set is an orthonormal set if we have

$$\langle\psi_j|\psi_k\rangle = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{if } j \neq k \end{cases} \tag{3}$$

for all choices of $j, k \in \{1, \ldots, m\}$.

Finally a set $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$ is an ***orthonormal*** basis if, in addition to being an orthonormal set, it forms a basis. This is equivalent to $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$ being an orthonormal set and m being equal to the dimension of the space from which $|\psi_1\rangle, \ldots, |\psi_m\rangle$ are drawn.

For example, for any classical state set Σ, the set of all standard basis vectors

$$\{|a\rangle : a \in \Sigma\}$$

is an ***orthonormal*** basis.

The set $\{|+\rangle, |-\rangle\}$ is an orthonormal basis for the 2-dimensional space corresponding to a single qubit, and the Bell basis $\{|\phi^+\rangle, |\phi^-\rangle, |\psi^+\rangle, |\psi^-\rangle\}$ is an orthonormal basis for the 4-dimensional space corresponding to two qubits.

Extending orthonormal sets to orthonormal bases : Suppose that $|\psi_1\rangle, \ldots, |\psi_m\rangle$ are vectors that live in an n-dimensional space, and assume moreover that $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$ is an orthonormal set.

Orthonormal sets are always linearly independent sets, so these vectors necessarily span a subspace of dimension m. From this we immediately conclude that $m \leq n$ because the dimension of the subspace spanned by these vectors cannot be larger than the dimension of the entire space from which they're drawn.

If it is the case that $m < n$, then it is always possible to choose an additional $n - m$ vectors $|\psi_{m+1}\rangle, \ldots, |\psi_n\rangle$ so that $\{|\psi_1\rangle, \ldots, |\psi_n\rangle\}$ forms an orthonormal basis. A procedure known as the Gram–Schmidt orthogonalization process can be used to construct these vectors.

The ***Gram-Schmidt*** process is a method used to orthogonalize a set of vectors. Given a set of linearly independent vectors, it constructs an orthonormal set from them. Here's how the process works:

1. **Start with the original set of vectors**: Let's denote these vectors as $v_1, v_2, ..., v_m$.

2. **Normalize the first vector**: Set the first vector of the new set to be the normalized version of the first vector from the original set: $u_1 = \frac{v_1}{|v_1|}$, where $|v_1|$ denotes the Euclidean norm of v_1.

3. **Orthogonalize the remaining vectors**: For each subsequent vector v_i from the original set, subtract the projection of v_i onto the vectors $u_1, u_2, ..., u_{i-1}$ from v_i itself. This ensures that each new vector is orthogonal to all previous vectors in the set.

   Mathematically, this step can be expressed as:

   $$u_i = v_i - \sum_{j=1}^{i-1} \text{proj}_{u_j}(v_i)$$

   where $\text{proj}_{u_j}(v_i)$ represents the projection of v_i onto u_j.

4. **Normalize the resulting vectors**: After obtaining the orthogonal vectors u_i, normalize each of them to obtain a set of orthonormal vectors.

   Mathematically, this step is expressed as:

   $$e_i = \frac{u_i}{|u_i|}$$

5. **Repeat steps 3 and 4 for all vectors**: Continue the process until all vectors have been orthogonalized and normalized.

At the end of this process, you will obtain an orthonormal set of vectors $e_1, e_2, ..., e_m$, which spans the same subspace as the original set of vectors but is orthogonal to each other. If the original set had fewer vectors than the dimension of the space, you can continue this process by adding more vectors to the set until you have a basis for the entire space.

Let's consider a simple example in a 3-dimensional space. Suppose we have two vectors v_1 and v_2 as follows:

$$v_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

We want to orthogonalize and normalize these vectors to obtain an orthonormal basis for the subspace they span.

1. **Normalize the first vector**:

   $$u_1 = \frac{v_1}{|v_1|} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix}$$

2. **Orthogonalize the second vector**:

   $$u_2 = v_2 - \text{proj}_{u_1}(v_2)$$

The projection of v_2 onto u_1 is:

$$\text{proj}_{u_1}(v_2) = \frac{v_2 \cdot u_1}{\|u_1\|^2} u_1$$

Calculate:

$$v_2 \cdot u_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}$$

So,

$$\text{proj}_{u_1}(v_2) = \frac{\frac{1}{\sqrt{2}}}{\frac{1}{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix}$$

Therefore:

$$u_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} - \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ \frac{-1}{2} \\ 1 \end{bmatrix}$$

3. **Normalize the resulting vectors**:

$$e_1 = \frac{u_1}{|u_1|} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \end{bmatrix}$$

$$e_2 = \frac{u_2}{|u_2|} = \frac{1}{\sqrt{\frac{1}{4} + \frac{1}{4} + 1}} \begin{bmatrix} \frac{1}{2} \\ \frac{-1}{2} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \\ 0 \end{bmatrix}$$

Now, e_1 and e_2 form an orthonormal basis for the subspace spanned by v_1 and v_2 in the 3-dimensional space.

Orthonormal sets and unitary matrices : Orthonormal sets of vectors are closely connected with unitary matrices. One way to express this connection is to say that the following three statements are logically equivalent (meaning that they are all true or all false) for any choice of a square matrix U:

1. The matrix U is ***Unitary*** (i.e., $U^\dagger U = \mathbb{I} = UU^\dagger$)
2. The rows of U form an ***Orthonormal set***.
3. The columns of U form an ***Orthonormal set***.

This equivalence is actually pretty straightforward when we think about how matrix multiplication and the conjugate transpose work. Suppose, for instance, that we have a 3×3 matrix like this:

$$U = \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} \\ \alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} \\ \alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3} \end{pmatrix}$$

The conjugate transpose of U looks like this:

$$U^\dagger = \begin{pmatrix} \overline{\alpha_{1,1}} & \overline{\alpha_{2,1}} & \overline{\alpha_{3,1}} \\ \overline{\alpha_{1,2}} & \overline{\alpha_{2,2}} & \overline{\alpha_{3,2}} \\ \overline{\alpha_{1,3}} & \overline{\alpha_{2,3}} & \overline{\alpha_{3,3}} \end{pmatrix}$$

Multiplying the two matrices, with the conjugate transpose on the left-hand side, gives us this matrix:

$$\begin{aligned} U \times U^\dagger &= \begin{pmatrix} \overline{\alpha_{1,1}} & \overline{\alpha_{2,1}} & \overline{\alpha_{3,1}} \\ \overline{\alpha_{1,2}} & \overline{\alpha_{2,2}} & \overline{\alpha_{3,2}} \\ \overline{\alpha_{1,3}} & \overline{\alpha_{2,3}} & \overline{\alpha_{3,3}} \end{pmatrix} \begin{pmatrix} \alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} \\ \alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} \\ \alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3} \end{pmatrix} \\ &= \begin{pmatrix} \overline{\alpha_{1,1}}\alpha_{1,1} + \overline{\alpha_{2,1}}\alpha_{2,1} + \overline{\alpha_{3,1}}\alpha_{3,1} & \overline{\alpha_{1,1}}\alpha_{1,2} + \overline{\alpha_{2,1}}\alpha_{2,2} + \overline{\alpha_{3,1}}\alpha_{3,2} & \overline{\alpha_{1,1}}\alpha_{1,3} + \overline{\alpha_{2,1}}\alpha_{2,3} + \overline{\alpha_{3,1}}\alpha_{3,3} \\ \overline{\alpha_{1,2}}\alpha_{1,1} + \overline{\alpha_{2,2}}\alpha_{2,1} + \overline{\alpha_{3,2}}\alpha_{3,1} & \overline{\alpha_{1,2}}\alpha_{1,2} + \overline{\alpha_{2,2}}\alpha_{2,2} + \overline{\alpha_{3,2}}\alpha_{3,2} & \overline{\alpha_{1,2}}\alpha_{1,3} + \overline{\alpha_{2,2}}\alpha_{2,3} + \overline{\alpha_{3,2}}\alpha_{3,3} \\ \overline{\alpha_{1,3}}\alpha_{1,1} + \overline{\alpha_{2,3}}\alpha_{2,1} + \overline{\alpha_{3,3}}\alpha_{3,1} & \overline{\alpha_{1,3}}\alpha_{1,2} + \overline{\alpha_{2,3}}\alpha_{2,2} + \overline{\alpha_{3,3}}\alpha_{3,2} & \overline{\alpha_{1,3}}\alpha_{1,3} + \overline{\alpha_{2,3}}\alpha_{2,3} + \overline{\alpha_{3,3}}\alpha_{3,3} \end{pmatrix} \end{aligned}$$

If we form three vectors from the columns of U,

$$|\psi_1\rangle = \begin{pmatrix} \alpha_{1,1} \\ \alpha_{2,1} \\ \alpha_{3,1} \end{pmatrix}, |\psi_2\rangle = \begin{pmatrix} \alpha_{1,2} \\ \alpha_{2,2} \\ \alpha_{3,2} \end{pmatrix}, |\psi_3\rangle = \begin{pmatrix} \alpha_{1,3} \\ \alpha_{2,3} \\ \alpha_{3,3} \end{pmatrix}$$

then we can alternatively express the product above as follows:

$$U^\dagger U = \begin{pmatrix} \langle\psi_1|\psi_1\rangle & \langle\psi_1|\psi_2\rangle & \langle\psi_1|\psi_3\rangle \\ \langle\psi_2|\psi_1\rangle & \langle\psi_2|\psi_2\rangle & \langle\psi_2|\psi_3\rangle \\ \langle\psi_3|\psi_1\rangle & \langle\psi_3|\psi_2\rangle & \langle\psi_3|\psi_3\rangle \end{pmatrix}$$

Referring to the equation (3), we now see that the condition that this matrix is equal to the identity matrix is equivalent to the orthonormality of the set $\{|\psi_1\rangle, |\psi_2\rangle, |\psi_3\rangle\}$.

This argument generalizes to unitary matrices of any size. The fact that the rows of a matrix form an orthonormal basis if and only if the matrix is unitary then follows from the fact that a matrix is unitary if and only if its transpose is unitary.

Given the equivalence described above, together with the fact that every orthonormal set can be extended to form an orthonormal basis, we conclude the following useful fact: Given any orthonormal set of vectors $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$ drawn from an n-dimensional space, there exists a unitary matrix U whose first m columns are the vectors $|\psi_1\rangle, \ldots, |\psi_m\rangle$.

Pictorially, we can always find a unitary matrix having this form:

$$U = \begin{pmatrix} | & | & & | & | & & | \\ |\psi_1\rangle & |\psi_2\rangle & \ldots & |\psi_m\rangle & |\psi_{m+1}\rangle & \ldots & |\psi_n\rangle \\ | & | & & | & | & & | \end{pmatrix}.$$

Here, the last $n-m$ columns are filled in with any choice of vectors $|\psi_{m+1}\rangle, \ldots, |\psi_n\rangle$ that make $\{|\psi_1\rangle, \ldots, |\psi_n\rangle\}$ an orthonormal basis.

Projections and projective measurements : Projection Matrices :

A square matrix Π is called a **projection** if it satisfies two properties:

1. $\Pi = \Pi^\dagger$
2. $\Pi^2 = \Pi$.

Matrices that satisfy the first condition — that they are equal to their own conjugate transpose — are called **Hermitian matrices**, and matrices that satisfy the second condition — that squaring them leaves them unchanged — are called **idempotent matrices**.

The word *projection* is sometimes used to refer to any matrix that satisfies just the **second condition but not necessarily the first**, and when this is done the term *orthogonal projection* is typically used to refer to matrices **satisfying both properties**. However, we will use the terms projection and projection matrix to mean matrices satisfying both conditions.

An example of a projection is the matrix

$$\Pi = |\psi\rangle\langle\psi| \tag{4}$$

for any unit vector $|\ \psi\rangle$. We can see that this matrix is Hermitian as follows:

$$\Pi^\dagger = (|\psi\rangle\langle\psi|)^\dagger = (\langle\psi|)^\dagger(|\psi\rangle)^\dagger = |\psi\rangle\langle\psi| = \Pi.$$

Here, to obtain the second equality, we have used the formula

$$(AB)^\dagger = B^\dagger A^\dagger,$$

which is always true (for any two matrices A and B for which the product AB makes sense).

To see that the matrix Π in (4) is idempotent, we can use the assumption that $|\ \psi\rangle$ is a unit vector, so that it satisfies $\langle\psi\ |\ \psi\rangle = 1$. Thus, we have

$$\Pi^2 = (|\psi\rangle\langle\psi|)^2 = |\psi\rangle\langle\psi|\psi\rangle\langle\psi| = |\psi\rangle\langle\psi| = \Pi$$

More generally, if $\{|psi_1\rangle, \ldots, |\psi_m\rangle\}$ is any orthonormal set of vectors, then the matrix

$$\Pi = \sum_{k=1}^{m} |\psi_k\rangle\langle\psi_k| \tag{5}$$

is a projection. Specifically, we have

$$\begin{aligned}
\Pi^\dagger &= \left(\sum_{k=1}^{m} |\psi_k\rangle\langle\psi_k|\right)^\dagger \\
&= \sum_{k=1}^{m} (|\psi_k\rangle\langle\psi_k|)^\dagger \\
&= \sum_{k=1}^{m} |\psi_k\rangle\langle\psi_k| \\
&= \Pi
\end{aligned}$$

and

$$\begin{aligned}\Pi^2 &= (\sum_{j=1}^{m} |\psi_j\rangle\langle\psi_j|)(\sum_{k=1}^{m} |\psi_k\rangle\langle\psi_k|) \\ &= \sum_{j=1}^{m}\sum_{k=1}^{m} |\psi_j\rangle\langle\psi_j|\psi_k\rangle\langle\psi_k| \\ &= \sum_{k=1}^{m} |\psi_k\rangle\langle\psi_k| \\ &= \Pi,\end{aligned}$$

where the orthonormality of $\{|psi_1\rangle, \ldots, |\psi_m\rangle\}$ is used just for the second-to-last equality.

In fact, this exhausts all of the possibilities: every projection Π can be written in the form (5) for some choice of an orthonormal set $\{|\psi_1\rangle, \ldots, |\psi_m\rangle\}$. (The zero matrix $\Pi = 0$, which is a projection, is a special case: to fit it into the general form (5) we have to allow the possibility that the sum is empty, resulting in the zero matrix.)

Projective Measurements :

As has already been mentioned, the notion of a measurement of a quantum system is more general than just standard basis measurements.

Projective measurements are measurements that are described by a collection of projections whose sum is equal to the identity matrix. In symbols, a collection $\{\Pi_1, \ldots, \Pi_{m-1}\}$ of projection matrices describes a projective measurement if

$$\Pi_0 + \ldots + \Pi_{m-1} = \mathbb{I},$$

When such a measurement is performed on a system X while it is in some state $|\ \psi\rangle$, two things happen:

1. For each $k \in \{0, \ldots, m-1\}$,the outcome of the measurement is k with probability equal to

$$Pr(\text{outcome is } k) = ||\Pi_k|\psi\rangle||^2.$$

2. For whichever outcome k the measurement produces, the state of X becomes

$$\frac{\Pi_k|\psi\rangle}{||\Pi_k|\psi\rangle||}$$

[ ]: