Intro to Quantum Computing

Prepared by Mark on April 3, 2024

Part 1: Probabilistic Bits

Definition 1:
As we already know, a *classical bit* may take the values `0` and `1`.
We can model this with a two-sided coin, one face of which is labeled `0`, and the other, `1`.

Of course, if we toss such a "bit-coin," we'll get either `0` or `1`.
We'll denote the probability of getting `0` as p_0, and the probability of getting `1` as p_1.
As with all probabilities, $p_0 + p_1$ must be equal to 1.

Definition 2:
Say we toss a "bit-coin" and don't observe the result. We now have a *probabilistic bit*, with a probability p_0 of being `0`, and a probability p_1 of being `1`.

We'll represent this probabilistic bit's *state* as a vector: $\left[\begin{smallmatrix} p_0 \\ p_1 \end{smallmatrix}\right]$
We do **not** assume this coin is fair, and thus p_0 might not equal p_1.
This may seem a bit redundant: since $p_0 + p_1$, we can always calculate one probability given the other.
We'll still include both probabilities in the state vector, since this provides a clearer analogy to quantum bits.

Definition 3:
The simplest probabilistic bit states are of course $[0]$ and $[1]$, defined as follows:

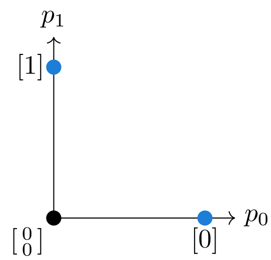
- $[0] = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$
- $[1] = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$

That is, $[0]$ represents a bit that we known to be `0`,
and $[1]$ represents a bit we know to be `1`.

Definition 4:
$[0]$ and $[1]$ form a *basis* for all possible probabilistic bit states:
Every other probabilistic bit can be written as a *linear combination* of $[0]$ and $[1]$:

$$\begin{bmatrix} p_0 \\ p_1 \end{bmatrix} = p_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + p_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = p_0[0] + p_1[1]$$

Problem 5:
Every possible state of a probabilistic bit is a two-dimensional vector.
Draw all possible states on the axis below.



Part 2: Measuring Probabilistic Bits

Definition 6:
As we noted before, a probabilistic bit represents a coin we've tossed but haven't looked at.
We do not know whether the bit is `0` or `1`, but we do know the probability of both of these outcomes.

If we *measure* (or *observe*) a probabilistic bit, we see either `0` or `1`—and thus our knowledge of its state is updated to either $[0]$ or $[1]$, since we now certainly know what face the coin landed on.

Since measurement changes what we know about a probabilistic bit, it changes the probabilistic bit's state. When we measure a bit, it's state *collapses* to either $[0]$ or $[1]$, and the original state of the bit vanishes. We *cannot* recover the state $[x_0, x_1]$ from a measured probabilistic bit.

Definition 7: Multiple bits
Say we have two probabilistic bits, x and y,
with states $[x] = [x_0, x_1]$ and $[y] = [y_0, y_1]$

The *compound state* of $[x]$ and $[y]$ is exactly what it sounds like:
It is the probabilistic two-bit state $|xy\rangle$, where the probabilities of the first bit are determined by $[x]$, and the probabilities of the second are determined by $[y]$.

Problem 8:
Say $[x] = [2/3, 1/3]$ and $[y] = [3/4, 1/4]$.

- If we measure x and y simultaneously,
  what is the probability of getting each of `00`, `01`, `10`, and `11`?
- If we measure y first and observe `1`,
  what is the probability of getting each of `00`, `01`, `10`, and `11`?

Note: $[x]$ and $[y]$ are column vectors, but I've written them horizontally to save space.

Problem 9:
Say $[x] = [2/3, 1/3]$ and $[y] = [3/4, 1/4]$.
What is the probability that x and y produce different outcomes?

Part 3: Tensor Products

Definition 10: Tensor Products
The *tensor product* of two vectors is defined as follows:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \otimes \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \\ x_2 \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} x_1 y_1 \\ x_1 y_2 \\ x_2 y_1 \\ x_2 y_2 \end{bmatrix}$$

That is, we take our first vector, multiply the second vector by each of its components, and stack the result. You could think of this as a generalization of scalar mulitiplication, where scalar mulitiplication is a tensor product with a vector in $\mathbb{R}^1$:

$$a \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_1 \end{bmatrix} \otimes \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_1 \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} a_1 y_1 \\ a_1 y_2 \end{bmatrix}$$

Problem 11:
Say $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$.
What is the dimension of $x \otimes y$?

Problem 12:
What is the pairwise tensor product $\left\{ \left[\begin{smallmatrix} 1 \\ 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 0 \\ 1 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 0 \\ 0 \\ 1 \end{smallmatrix}\right] \right\} \otimes \left\{ \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right] \right\}$?
in other words, distribute the tensor product between every pair of vectors.

Problem 13:
What is the *span* of the vectors we found in Problem 12?
In other words, what is the set of vectors that can be written as linear combinations of the vectors above?

Problem 14:
Say $[x] = [2/3, 1/3]$ and $[y] = [3/4, 1/4]$.
What is $[x] \otimes [y]$? How does this relate to Problem 8?

Problem 15:
The compound state of two vector-form bits is their tensor product.
Compute the following. Is the result what we'd expect?

- $[0] \otimes [0]$
- $[0] \otimes [1]$
- $[1] \otimes [0]$
- $[1] \otimes [1]$

Hint: Remember that $[0] = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$ and $[1] = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$.

Problem 16:
Of course, writing $[0] \otimes [1]$ is a bit excessive. We'll shorten this notation to $[01]$.

In fact, we could go further: if we wanted to write the set of bits $[1] \otimes [1] \otimes [0] \otimes [1]$, we could write $[1101]$—but a shorter alternative is $[13]$, since 13 is `1101` in binary.

Write $[5]$ as three-bit probabilistic state.

Problem 17:
Write the three-bit states $[0]$ through $[7]$ as column vectors.
Hint: You do not need to compute every tensor product. Do a few and find the pattern.

Part 4: Operations on Probabilistic Bits

Now that we can write probabilistic bits as vectors, we can represent operations on these bits with linear transformations—in other words, as matrices.

Definition 18:
Consider the NOT gate, which operates as follows:

- NOT[0] = [1]
- NOT[1] = [0]

What should NOT do to a probabilistic bit $[x_0, x_1]$?
If we return to our coin analogy, we can think of the NOT operation as flipping a coin we have already tossed, without looking at its state. Thus,

$$\text{NOT} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_0 \end{bmatrix}$$

Review: Multiplying Vectors by Matrices

$$Av = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} v_0 \\ v_1 \end{bmatrix} = \begin{bmatrix} 1v_0 + 2v_1 \\ 3v_0 + 4v_1 \end{bmatrix}$$

Note that each element of Av is the dot product of a row in A and a column in v.

Problem 19:
Compute the following product:

$$\begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Remark:
Also, recall that every matrix is linear map, and that every linear map may be written as a matrix. We often use the terms *matrix*, *transformation*, and *linear map* interchangably.

Problem 20:
Find the matrix that represents the NOT operation on one probabilistic bit.

Problem 21: Extension by linearity
Say we have an arbitrary operation M.
If we know how M acts on [1] and [0], can we compute $M[x]$ for an arbitrary state $[x]$?
Say $[x] = [x_0, x_1]$.

- What is the probability we observe 0 when we measure x?
- What is the probability that we observe $M[0]$ when we measure Mx?

Problem 22:
Write $M[x_0, x_1]$ in terms of $M[0]$, $M[1]$, x_0, and x_1.

Remark:
Every matrix represents a *linear* map, so the following is always true:

$$A \times (px + qy) = pAx + qAy$$

Problem 22 is just a special case of this fact.

Part 5: One Qubit

Quantum bits (or *qubits*) are very similar to probabilistic bits, but have one major difference: probabilities are replaced with *amplitudes*.

Of course, a qubit can take the values `0` and `1`, which are denoted $|0\rangle$ and $|1\rangle$.
Like probabilistic bits, a quantum bit is written as a linear combination of $|0\rangle$ and $|1\rangle$:

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle$$

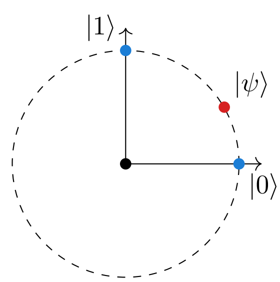
Such linear combinations are called *superpositions*.

The $|\ \rangle$ you see in the expressions above is called a "ket," and denotes a column vector.
$|0\rangle$ is pronounced "ket zero," and $|1\rangle$ is pronounced "ket one." This is called bra-ket notation.
Note: $\langle 0|$ is called a "bra," but we won't worry about that for now.

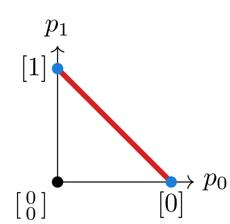
This is very similiar to the "box" $[\ ]$ notation we used for probabilistic bits.
As before, we will write $|0\rangle = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$ and $|1\rangle = \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]$.

Recall that probabilistic bits are subject to the restriction that $p_0 + p_1 = 1$.
Quantum bits have a similar condition: $\psi_0^2 + \psi_1^2 = 1$.
Note that this implies that ψ_0 and ψ_1 are both in $[-1, 1]$.
Quantum amplitudes may be negative, but probabilistic bit probabilities cannot.

If we plot the set of valid quantum states on our plane, we get a unit circle centered at the origin:



Recall that the set of probabilistic bits forms a line instead:



Problem 23:
In the above unit circle, the counterclockwise angle from $|0\rangle$ to $|\psi\rangle$ is 30°.
Write $|\psi\rangle$ as a linear combination of $|0\rangle$ and $|1\rangle$.

Definition 24: Measurement I
Just like a probabilistic bit, we must observed $|0\rangle$ or $|1\rangle$ when we measure a qubit.
If we were to measure $|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle$, we'd observe either $|0\rangle$ or $|1\rangle$,
with the following probabilities:

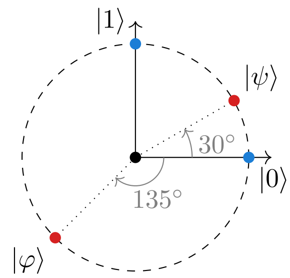
- $\mathcal{P}(|1\rangle) = \psi_1^2$
- $\mathcal{P}(|0\rangle) = \psi_0^2$

Note that $\mathcal{P}(|0\rangle) + \mathcal{P}(|1\rangle) = 1$.

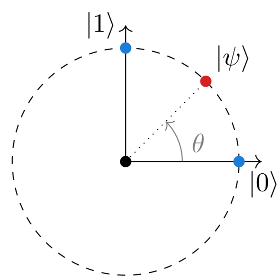
As before, $|\psi\rangle$ *collapses* when it is measured: its state becomes that which we observed in our measurement, leaving no trace of the previous superposition.

Problem 25:

- What is the probability we observe $|0\rangle$ when we measure $|\psi\rangle$?
- What can we observe if we measure $|\psi\rangle$ a second time?
- What are these probabilities for $|\varphi\rangle$?



As you may have noticed, we don't need two coordinates to fully define a quibit's state.
We can get by with one coordinate just as well.
Instead of referring to each state using its cartesian coordinates ψ_0 and ψ_1,
we can address it using its *polar angle* θ, measured from $|0\rangle$ counterclockwise:



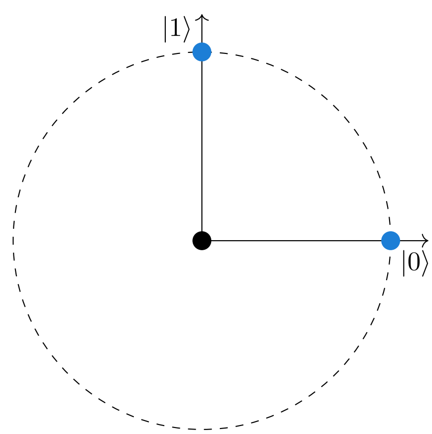
Problem 26:
Find ψ_0 and ψ_1 in terms of θ for an arbitrary qubit ψ.

Problem 27:

Consider the following qubit states:

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

- Where are these on the unit circle?
- What are their polar angles?
- What are the probabilities of observing $|0\rangle$ and $|1\rangle$ when measuring $|+\rangle$ and $|-\rangle$?



Part 6: Operations on One Qubit

We may apply transformations to qubits just as we apply transformations to probabilistic bits. Again, we'll represent transformations as 2×2 matrices, since we want to map one qubit state to another.
In other words, we want to map elements of $\mathbb{R}^2$ to elements of $\mathbb{R}^2$.
We will call such maps *quantum gates,* since they are the quantum equivalent of classical logic gates.

There are two conditions a valid quantum gate G must satisfy:

- For any valid state $|\psi\rangle$, $G\,|\psi\rangle$ is a valid state.
  Namely, G must preserve the length of any vector it is applied to.
  Recall that the set of valid quantum states is the set of unit vectors in $\mathbb{R}^2$
- Any quantum gate must be *invertible.*
  We'll skip this condition for now, and return to it later.

In short, a quantum gate is a linear map that maps the unit circle to itself.
There are only two kinds of linear maps that do this: reflections and rotations.

Problem 28:
The X gate is the quantum analog of the `not` gate, defined by the following table:

- $X\,|0\rangle = |1\rangle$
- $X\,|1\rangle = |0\rangle$

Find the matrix X.

Problem 29:
What is $X\,|+\rangle$ and $X\,|-\rangle$?
Hint: Remember that all matrices are linear maps. What does this mean?

Problem 30:
In terms of geometric transformations, what does X do to the unit circle?

Problem 31:

Let Z be a quantum gate defined by the following table:

- $Z|0\rangle = |0\rangle$,
- $Z|1\rangle = -|1\rangle$.

What is the matrix Z? What are $Z|+\rangle$ and $Z|-\rangle$?
What is Z as a geometric transformation?

Problem 32:

Is the map B defined by the table below a valid quantum gate?

- $B|0\rangle = |0\rangle$
- $B|1\rangle = |+\rangle$

Hint: Find a $|\psi\rangle$ so that $B|\psi\rangle$ is not a valid qubit state

Problem 33: Rotation

As we noted earlier, any rotation about the center is a valid quantum gate.
Let's derive all transformations of this form.

- Let U_ϕ be the matrix that represents a counterclockwise rotation of ϕ degrees.
  What is $U|0\rangle$ and $U|1\rangle$?
- Find the matrix U_ϕ for an arbitrary ϕ.

Problem 34:

Say we have a qubit that is either $|+\rangle$ or $|-\rangle$. We do not know which of the two states it is in.
Using one operation and one measurement, how can we find out, for certain, which qubit we received?

Part 7: Two Qubits

Definition 35:
Just as before, we'll represent multi-qubit states as linear combinations of multi-qubit basis states.
For example, a two-qubit state $|ab\rangle$ is the four-dimensional unit vector

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = a\,|00\rangle + b\,|01\rangle + c\,|10\rangle + d\,|11\rangle \tag{1}$$

As always, multi-qubit states are unit vectors.
Thus, $a^2 + b^2 + c^2 + d^2 = 1$ in the two-bit case above.

Problem 36:
Say we have two qubits $|\psi\rangle$ and $|\varphi\rangle$.
Show that $|\psi\rangle \otimes |\varphi\rangle$ is always a unit vector (and is thus a valid quantum state).

Definition 37: Measurement II
Measurement of a two-qubit state works just like measurement of a one-qubit state:
If we measure $a\,|00\rangle + b\,|01\rangle + c\,|10\rangle + d\,|11\rangle$,
we get one of the four basis states with the following probabilities:

- $\mathcal{P}(|00\rangle) = a^2$
- $\mathcal{P}(|01\rangle) = b^2$
- $\mathcal{P}(|10\rangle) = c^2$
- $\mathcal{P}(|11\rangle) = d^2$

As before, the sum of all the above probabilities is 1.

Problem 38:
Consider the two-qubit state $|\psi\rangle = \frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{2}\,|01\rangle + \frac{\sqrt{3}}{4}\,|10\rangle + \frac{1}{4}\,|11\rangle$

- If we measure both bits of $|\psi\rangle$ simultaneously,
  what is the probability of getting each of $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$?
- If we measure the ONLY the first qubit, what is the probability we get $|0\rangle$? How about $|1\rangle$?
  Hint: There are two basis states in which the first qubit is $|0\rangle$.
- Say we measured the second bit and read $|1\rangle$.
  If we now measure the first bit, what is the probability of getting $|0\rangle$?

Problem 39:
Again, consider the two-qubit state $|\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{2}|01\rangle + \frac{\sqrt{3}}{4}|10\rangle + \frac{1}{4}|11\rangle$
If we measure the first qubit of $|\psi\rangle$ and get $|0\rangle$, what is the resulting state of $|\psi\rangle$?
What would the state be if we'd measured $|1\rangle$ instead?

Problem 40:
Consider the three-qubit state $|\psi\rangle = c_0|000\rangle + c_1|001\rangle + ... + c_7|111\rangle$.
Say we measure the first two qubits and get $|00\rangle$. What is the resulting state of $|\psi\rangle$?

Definition 41: Entanglement
Some product states can be factored into a tensor product of individual qubit states. For example,

$$\frac{1}{2}\big(|00\rangle + |01\rangle + |10\rangle + |11\rangle\big) = \frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big) \otimes \frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big)$$

Such states are called *product states.* States that aren't product states are called *entangled* states.

Problem 42:
Factor the following product state:

$$\frac{1}{2\sqrt{2}}\big(\sqrt{3}\,|00\rangle - \sqrt{3}\,|01\rangle + |10\rangle - |11\rangle\big)$$

Problem 43:
Show that the following is an entangled state.

$$\frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle$$

Part 8: Logic Gates

Definition 44: Matrices
Throughout this handout, we've been using matrices. Again, recall that every linear map may be written as a matrix, and that every matrix represents a linear map. For example, if $f : \mathbb{R}^2 \to \mathbb{R}^2$ is a linear map, we can write it as follows:

$$f(|x\rangle) = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} m_1x_1 + m_2x_2 \\ m_3x_1 + m_4x_2 \end{bmatrix}$$

Definition 45:
Before we discussing multi-qubit quantum gates, we need to review to classical logic.
Of course, a classical logic gate is a linear map from $\{0,1\}^m$ to $\{0,1\}^n$

Problem 46:
The `not` gate is a map defined by the following table:

- $X\,|0\rangle = |1\rangle$
- $X\,|1\rangle = |0\rangle$

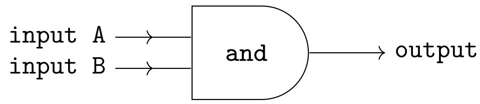
Write the `not` gate as a matrix that operates on single-bit vector states.
That is, find a matrix X so that $X\left[\begin{smallmatrix}1\\0\end{smallmatrix}\right] = \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]$ and $X\left[\begin{smallmatrix}0\\1\end{smallmatrix}\right] = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$

Problem 47:
The `and` gate is a map $\mathbb{B}^2 \to \mathbb{B}$ defined by the following table:

`a`	`b`	`a` and `b`
0	0	0
0	1	0
1	0	0
1	1	1

Find a matrix A so that $A\,|\texttt{ab}\rangle$ works as expected.
Hint: Remember, we write bits as vectors.

Remark:
The way a quantum circuit handles information is a bit different than the way a classical circuit does. We usually think of logic gates as *functions*: they consume one set of bits, and return another:

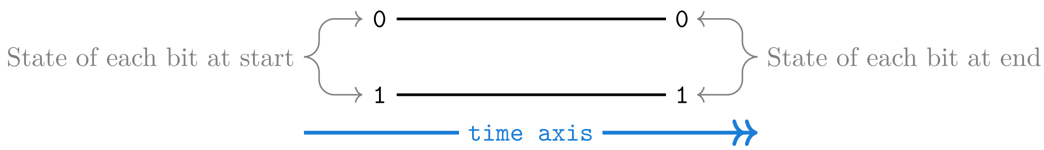


This model, however, won't work for quantum logic. If we want to understand quantum gates, we need to see them not as *functions*, but as *transformations*. This distinction is subtle, but significant:

- functions *consume* a set of inputs and *produce* a set of outputs
- transformations *change* a set of objects, without adding or removing any elements

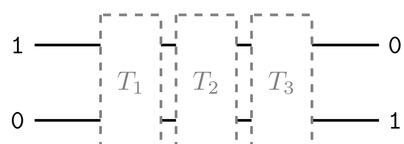
Our usual logic circuit notation models logic gates as functions—we thus can't use it.
We'll need a different diagram to draw quantum circuits.

First, we'll need a set of bits. For this example, we'll use two, drawn in a vertical array.
We'll also add a horizontal time axis, moving from left to right:



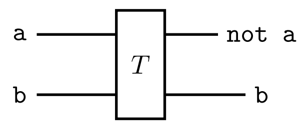
In the diagram above, we didn't change our bits—so the labels at the start match those at the end.

Thus, our circuit forms a grid, with bits ordered vertically and time horizontally.
If we want to change our state, we draw transformations as vertical boxes.
Every column represents a single transformation on the entire state:



Note that the transformations above span the whole state. This is important:
we cannot apply transformations to individual bits—we always transform the *entire* state.

Setup:
Say we want to invert the first bit of a two-bit state. That is, we want a transformation T so that

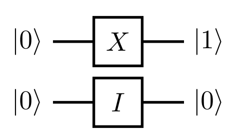


In other words, we want a matrix T satisfying the following equalities:

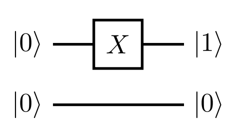
- $T\,|00\rangle = |10\rangle$
- $T\,|01\rangle = |11\rangle$
- $T\,|10\rangle = |00\rangle$
- $T\,|11\rangle = |01\rangle$

Problem 48:
Find the matrix that corresponds to the above transformation.
Hint: Remember that $|0\rangle = \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right]$ and $|1\rangle = \left[\begin{smallmatrix}0\\1\end{smallmatrix}\right]$
Also, we found earlier that $X = \left[\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right]$, and of course $I = \left[\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right]$.

Remark:
We could draw the above transformation as a combination X and I (identity) gate:



We can even omit the I gate, since we now know that transformations affect the whole state:



We're now done: this is how we draw quantum circuits. Don't forget that transformations *always* affect the whole state—even if our diagram doesn't explicitly state this.

Part 9: Quantum Gates

In the previous section, we stated that a quantum gate is a linear map.
Let's complete that definition.

Definition 49:
A quantum gate is a *orthonormal matrix*, which means any gate G satisfies $GG^{\mathrm{T}} = I$.
This implies the following:

- G is square. In other words, it has as many rows as it has columns.
  If we think of G as a map, this means that G has as many inputs as it has outputs.
  This is to be expected: we stated earlier that quantum gates do not destroy or create qubits.
- G preserves lengths; i.e $|x| = |Gx|$.
  This ensures that $G|\psi\rangle$ is always a valid state.

(You will prove all these properties in any introductory linear algebra course.
This isn't a lesson on linear algebra, so you may take them as given today.)

Remark:
Let G be a quantum gate.
Since quantum gates are, by definition, *linear* maps, the following holds:

$$G\big(a_0|0\rangle + a_1|1\rangle\big) = a_0 G|0\rangle + a_1 G|1\rangle$$

Problem 50:
Consider the *controlled not* (or *cnot*) gate, defined by the following table:

- $\mathrm{X_c}|00\rangle = |00\rangle$
- $\mathrm{X_c}|01\rangle = |01\rangle$
- $\mathrm{X_c}|10\rangle = |11\rangle$
- $\mathrm{X_c}|11\rangle = |10\rangle$

In other words, the cnot gate inverts its second bit if its first bit is $|1\rangle$.
Find the matrix that applies the cnot gate.

Problem 51:
Evaluate the following:

$$\mathrm{X_C}\Big(\frac{1}{2}\left|00\right\rangle + \frac{1}{2}\left|01\right\rangle - \frac{1}{2}\left|10\right\rangle - \frac{1}{2}\left|11\right\rangle\Big)$$

Problem 52:
If we measure the result of Problem 51, what are the probabilities of getting each state?

Problem 53:
Finally, modify the original cnot gate so that the roles of its bits are reversed:
$\mathrm{X}_{\text{c, flipped}} \left|ab\right\rangle$ should invert $\left|a\right\rangle$ iff $\left|b\right\rangle$ is $\left|1\right\rangle$.

Definition 54:
The *Hadamard Gate* is given by the following matrix:

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

Note that we divide by $\sqrt{2}$, since H must be orthonormal.

Review: Matrix Multiplication

Matrix multiplication works as follows:

$$AB = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} a_0 & b_0 \\ a_1 & b_1 \end{bmatrix} = \begin{bmatrix} 1a_0 + 2a_1 & 1b_0 + 2b_1 \\ 3a_0 + 4a_1 & 3b_0 + 4b_1 \end{bmatrix}$$

Note that this is very similar to multiplying each column of B by A.
The product AB is simply Ac for every column c in B:

$$Ac_0 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 1a_0 + 2a_1 \\ 3a_0 + 4a_1 \end{bmatrix}$$

This is exactly the first column of the matrix product.
Also, note that each element of Ac_0 is the dot product of a row in A and a column in c_0.

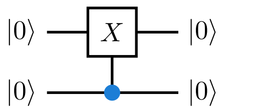
Problem 55:
What is HH?
Using this result, find H^{-1}.

Problem 56:
What geometric transformation does H apply to the unit circle?
Hint: Rotation or reflection? How much, or about which axis?

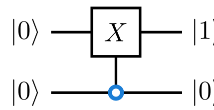
Part 10: HXH

Let's return to the quantum circuit diagrams we discussed a few pages ago.
Keep in mind that we're working with quantum gates and proper half-qubits—not classical bits, as we were before.

Definition 57: Controlled Inputs
A *control input* or *inverted control input* may be attached to any gate.
These are drawn as filled and empty circles in our circuit diagrams:



Non-inverted control input



Inverted control input

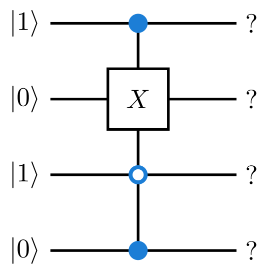
An X gate with a (non-inverted) control input behaves like an X gate if *all* its control inputs are $|1\rangle$, and like I otherwise. An X gate with an inverted control inputs does the opposite, behaving like I if its input is $|1\rangle$ and like X otherwise. The two circuits above illustrate this fact—take a look at their inputs and outputs.

Of course, we can give a gate multiple controls.
An X gate with multiplie controls behaves like an X gate if...

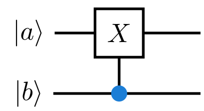
- all non-inverted controls are $|1\rangle$, and
- all inverted controls are $|0\rangle$

...and like I otherwise.

Problem 58:
What are the final states of the qubits in the diagram below?



Problem 59:
Consider the diagram below, with one controlled X gate:
Note: The CNOT gate from Problem 50 is a controlled X gate.



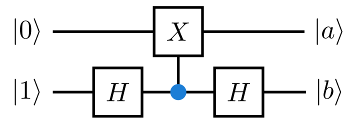
Find a matrix $\mathrm{X_c}$ that represents this gate, so that $\mathrm{X_c} |ab\rangle$ works as expected.

Problem 60:
Now, evaluate the following. Remember that $|+\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big)$ and $|-\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle - |1\rangle\Big)$

Hint: Note that some of these states are entangled. The circuit diagrams are a bit misleading: we can't write an entangled state as two distinct qubits!

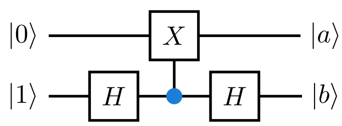
So, don't try to find $|a\rangle$ and $|b\rangle$.
Instead find $|ab\rangle = \psi_0 |00\rangle + \psi_1 |01\rangle + \psi_2 |10\rangle + \psi_3 |11\rangle$, and factor it into $|a\rangle \otimes |b\rangle$ if you can.

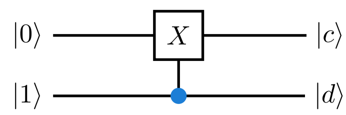
Remark:
Now, consider the following circuit:



We already know that H is its own inverse: $HH = I$.
Applying H to a qubit twice does not change its state.
Recall that $H = \frac{1}{\sqrt{2}}\left[\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix}\right]$

So, we might expect that the two circuits below are equivalent:
After all, we H the second bit, use it to control an X gate, and then H it back to its previous state.

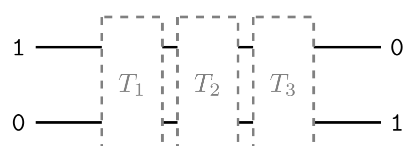




This, however, isn't the case:
If we compute the state $|ab\rangle$ in the left circuit, we get $[0.5,\ 0.5, -0.5,\ 0.5]$ (which is entangled),
but the state $|cd\rangle$ on the right is $|11\rangle = [0, 0, 0, 1]$.
This is easy to verify with a few matrix multiplications.

How does this make sense?
Remember that a two-bit quantum state is *not* equivalent to a pair of one-qubit quantum states. We must treat a multi-qubit state as a single unit.
Recall that a two-bit state $|ab\rangle$ comes with four probabilities: $\mathcal{P}(\texttt{00})$, $\mathcal{P}(\texttt{01})$, $\mathcal{P}(\texttt{10})$, and $\mathcal{P}(\texttt{11})$. If we change the probabilities of only $|a\rangle$, *all four of these change!*

Because of this fact, "controlled gates" may not work as you expect. They may seem to "read" their controlling qubit without affecting its state, but remember—a controlled gate still affects the *entire* state. As we noted before, it is not possible to apply a transformation to one bit of a quantum state.



Part 11: Superdense Coding

Consider the following entangled two-qubit states, called the *bell states*:

- $|\Phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$
- $|\Phi^-\rangle = \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|11\rangle$
- $|\Psi^+\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle$
- $|\Psi^-\rangle = \frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|10\rangle$

Problem 61:
The probabilistic bits we get when measuring any of the above may be called *anticorrelated bits*.
If we measure the first bit of any of these states and observe 1, what is the resulting compound state?
What if we observe 0 instead?
Do you see why we can call these bits anticorrelated?

Problem 62:
Show that the bell states are orthogonal
Hint: Dot product

Problem 63:
Say we have a pair of qubits in one of the four bell states.
How can we find out which of the four states we have, with certainty?
Hint: $H|+\rangle = |0\rangle$, and $H|-\rangle = |1\rangle$

Definition 64:
The Z gate is defined as follows:

$$Z \begin{bmatrix} \psi_0 \\ \psi_1 \end{bmatrix} = \begin{bmatrix} \psi_0 \\ -\psi_1 \end{bmatrix}$$

Problem 65:
Suppose that Alice and Bob are each in possession of one qubit.
These two qubits are entangled, and have the compound state $|\Phi^+\rangle$.
Note: We could say that they each have "half" of $|\Phi^+\rangle$.
How can Alice send a two-bit classical state (i.e, one of the four values `00`, `01`, `10`, `11`) to Bob by only sending one qubit?

Remark:
Superdense coding consumes a pre-shared entangled pair to transmit two bits of information. This entanglement may *not* be re-used—it is destroyed when Bob measures the final qubit states.

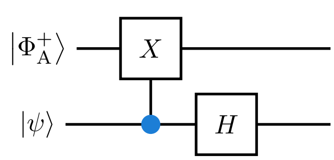
Part 12: Quantum Teleportation

Superdense coding lets us convert quantum bandwidth into classical bandwidth.
Quantum teleporation does the opposite, using two classical bits and an entangled pair to transmit a quantum state.

Setup:
Again, suppose Alice and Bob each have half of a $|\Phi^+\rangle$ state.
We'll call the state Alice wants to teleport $|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle$.

Problem 66:
What is the three-qubit state $|\psi\rangle |\Phi^+\rangle$ in terms of ψ_0 and ψ_1?

Problem 67:
To teleport $|\psi\rangle$, Alice applies the following circuit to her two qubits, where $\left|\Phi^+_\mathrm{A}\right\rangle$ is her half of $|\Phi^+\rangle$. She then measures both qubits and sends the result to Bob.



What should Bob do so that $\left|\Phi^+_B\right\rangle$ takes the state $|\psi\rangle$ had initially?

Problem 68:
With an informal proof, show that it is not possible to use superdense coding to send more than two classical bits through an entangled two-qubit quantum state.