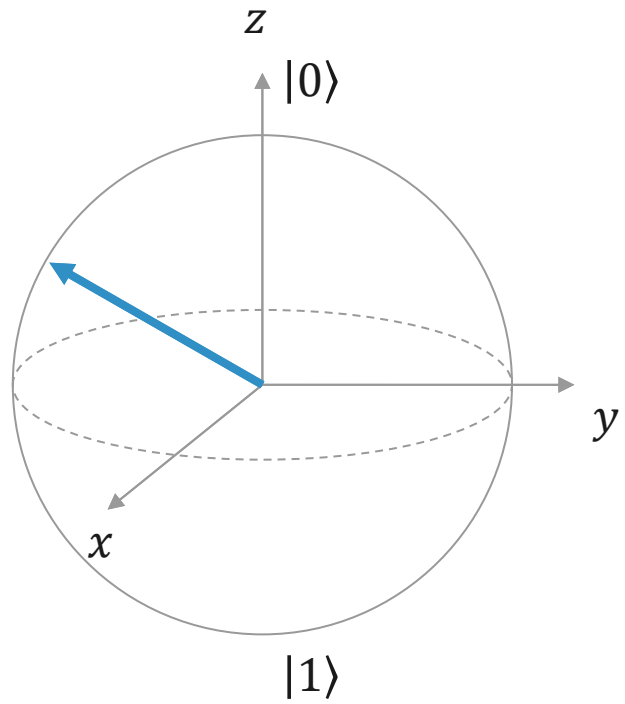
SAJID Lab
Smart & Advanced Junction of Intelligent Devices

EEE 6516 · QUANTUM COMPUTING

Why Quantum Computing?

Complex vectors and Dirac notation



Chapter 1 · Lecture 1

Sajid Muhaimin Choudhury

Copyright © 2026 Sajid Muhaimin Choudhury. All rights reserved.

Today

1. **Why quantum computing**
2. **Bits to qubits**
3. **Complex numbers**
4. **Kets and bras**
5. **Three simulators**

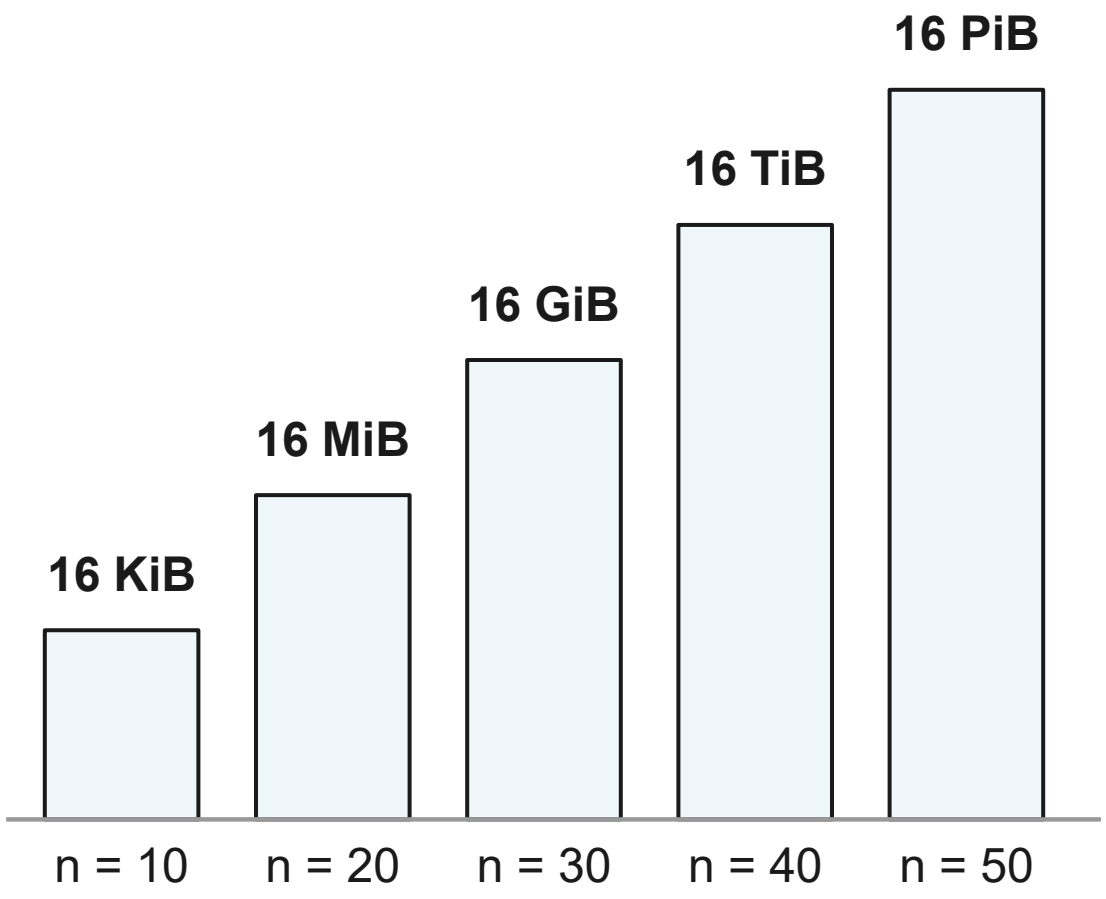
By the end of this lecture you can

- explain why n qubits need 2^n complex numbers
- contrast probability vectors with amplitude vectors
- write kets and bras; normalise $\alpha|0\rangle + \beta|1\rangle$
- create the same state in NumPy, Qiskit and QuTiP

1 Why quantum computing?

What does it cost to describe n qubits on an ordinary computer?

Each added qubit doubles the description



A register of n qubits is described by 2^n complex amplitudes.

One more qubit: twice as many numbers.

Bar heights are logarithmic: every bar is 1024 times the previous one.

WORKED EXAMPLE · STEP 1 OF 2

Worked example: the memory wall

$$\text{memory} = 16 \cdot 2^n \text{ bytes} = 2^{n+4} \text{ bytes}$$

$n = 30$: 2^{34} bytes $= 16$ GiB

1 16 bytes per amplitude, 2^{30} amplitudes: 16 GiB (a laptop).

Before the next slide

How much for 50 qubits?

Worked example: the memory wall

$$\text{memory} = 16 \cdot 2^n \text{ bytes} = 2^{n+4} \text{ bytes}$$

$n = 30$: 2^{34} bytes $= 16$ GiB

$n = 50$: 2^{54} bytes $= 16$ PiB $\approx 1.8 \times 10^{16}$ B

1 16 bytes per amplitude, 2^{30} amplitudes: 16 GiB (a laptop).

2 50 qubits: 2^{54} bytes, beyond any single machine. Twenty more qubits multiply the memory by about a million.

SIMULATION

The same arithmetic in Python

```
BYTES = 16
for n in range(10, 51, 10):
    amps = 2 ** n
    size = human(BYTES * amps)
```

human() converts bytes to KiB, MiB, ... (binary prefixes, 1 KiB = 1024 B).

Recorded output on the right.

n	memory
10	16 KiB
20	16 MiB
30	16 GiB
40	16 TiB
50	16 PiB

COMMON MISTAKE

Recap from Lecture 0: “It tries every answer at once”

✗ 2^n parallel computations

2^n amplitudes = 2^n computations running side by side;
read off the best answer at the end.

Reading out n qubits gives only n bits, and a careless readout is random.

✓ Interference by design

The algorithm arranges the amplitudes so that wrong answers cancel and the right one becomes likely.

Cancellation, used deliberately (next section).

A quantum speed-up comes from interference, not from reading many answers.

2 From bits to qubits

What changes when the numbers describing a bit become complex?

Three one-bit registers, one kind of vector

Bit		Probabilistic bit		Qubit
$\begin{bmatrix}1\\0\end{bmatrix}$	$\longrightarrow$	$\begin{bmatrix}p_0\\p_1\end{bmatrix}$	$\longrightarrow$	$\begin{bmatrix}\alpha\\\beta\end{bmatrix}$
one entry is 1		$p_0, p_1 \geq 0$ $p_0 + p_1 = 1$		α, β complex $\lvert\alpha\rvert^2 + \lvert\beta\rvert^2 = 1$
		entries only add		*entries can cancel*

Only the rules for the two entries change.

A random flip: probabilities only add

$$F = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$F\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix}, \qquad F\begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix}$$

Columns of a random operation are probability vectors.

Two random flips leave the bit as random as one.

Once randomness is in, no random operation removes it.

YOUR TURN · CALCULATE

Lecture 0's two beam splitters, in matrices

Work it out · 1 minute, then vote

The quantum analogue of the flip is $H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

Start from $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and apply H twice. Multiply it out.

A still an equal mix

B back to (1, 0)

C (0, 1)

D it depends on chance

ANSWER

B: the second application cancels

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

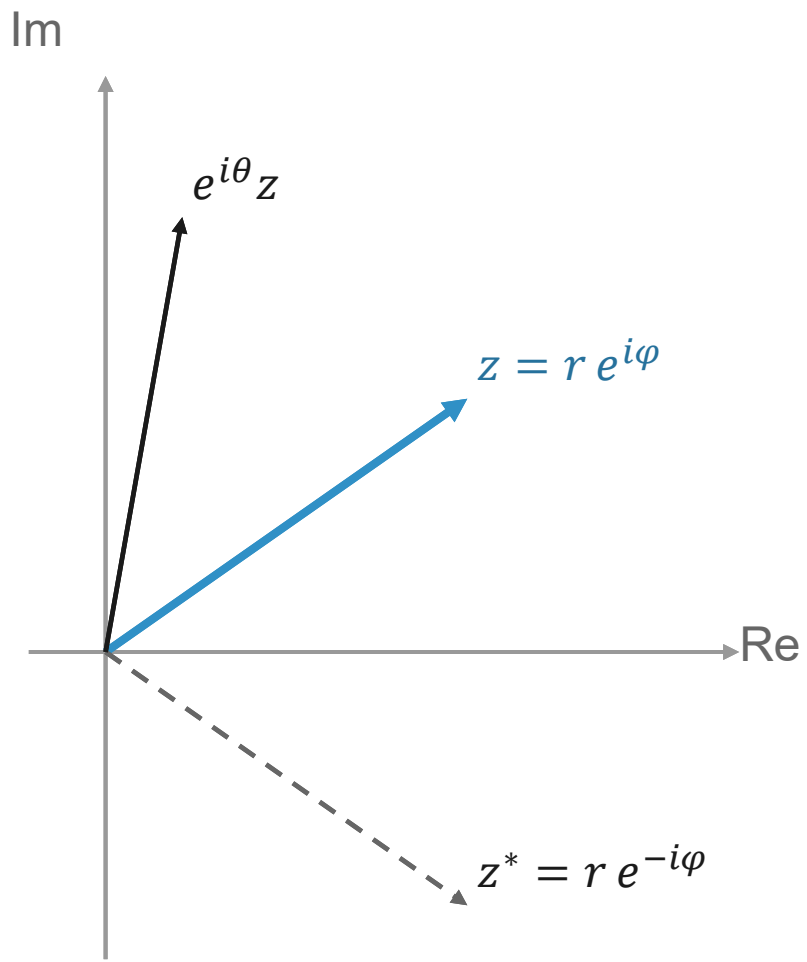
$$H\begin{bmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{bmatrix} = \begin{bmatrix}\frac{1}{2}+\frac{1}{2}\\ \frac{1}{2}-\frac{1}{2}\end{bmatrix} = \begin{bmatrix}1\\0\end{bmatrix}$$

Negative (complex) entries let contributions cancel: the resource every quantum algorithm uses.

3 Complex numbers revisited

You already use them every day as phasors.

An amplitude is a phasor



$$z = a + ib = |z|\,e^{i\varphi}$$

$$z^* = a - ib, \qquad z^*z = |z|^2 = a^2 + b^2$$

Multiplying by $e^{i\theta}$ rotates; the modulus is unchanged.

This course writes i where circuit texts write j.

EE LINK

Phasors and amplitudes: where the analogy stops

Same: adding two phasors predicts reinforcement or cancellation; adding amplitudes does the same.

Different: a phasor describes a deterministic waveform;

the squared modulus $|\alpha|^2$ of an amplitude is a probability.

YOUR TURN · CALCULATE

Checkpoint: modulus and phase of a product

Work it out · 2 minutes

For $z = 1 + i$ and $w = \sqrt{3} - i$, find $|zw|$ and the phase of zw without multiplying out.

4 Kets, bras and normalisation

A compact name for column and row vectors.

Kets are columns, bras are conjugate rows

Definition · ket and computational basis

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

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

Definition · bra

$$\langle\psi| = [\alpha^* \quad \beta^*]$$

Normalised: $|\alpha|^2 + |\beta|^2 = 1$

Only normalised kets describe qubits.

Normalise by dividing by $\sqrt{|\alpha|^2 + |\beta|^2}$.

WORKED EXAMPLE · STEP 1 OF 3

Worked example: normalising a state

$$|\chi\rangle = (1 + i)|0\rangle + 2|1\rangle$$

1 Write down the unnormalised ket.

Before the next slide

What are the two squared moduli?

Worked example: normalising a state

$$|\chi\rangle = (1+i)|0\rangle + 2|1\rangle$$

$$|1+i|^2 = 2, \qquad |2|^2 = 4, \qquad \||\chi\rangle\| = \sqrt{6}$$

1 Write down the unnormalised ket.

2 Squared moduli, not squares: $|1+i|^2 = 1^2 + 1^2$.

Before the next slide

Divide by what?

Worked example: normalising a state

$$|\chi\rangle = (1+i)|0\rangle + 2|1\rangle$$

$$|1+i|^2 = 2, \qquad |2|^2 = 4, \qquad \||\chi\rangle\| = \sqrt{6}$$

$$|\psi\rangle = \frac{1+i}{\sqrt{6}}|0\rangle + \frac{2}{\sqrt{6}}|1\rangle$$

1 Write down the unnormalised ket.

2 Squared moduli, not squares: $|1+i|^2 = 1^2 + 1^2$.

3 Divide by $\sqrt{6}$: $|\alpha|^2 = \frac{1}{3}, |\beta|^2 = \frac{2}{3}$

COMMON MISTAKE

Squared modulus, not square

✗ Squaring the amplitude

$$(1+i)^2 + 2^2 = 4 + 2i$$

Not even a real number.

✓ Squared modulus

$$|1+i|^2 = (1-i)(1+i) = 2$$

$$2 + 4 = 6$$

Real and non-negative.

Use $|\alpha|^2 = \alpha^* \alpha$. Forgetting the conjugate in a bra is the same slip.

YOUR TURN · APPLY

Your turn: write the bra

Think · 90 seconds

Write $\langle\psi|$ for $|\psi\rangle = \frac{1}{5}(3|0\rangle + 4i|1\rangle)$.

Is the state normalised?

ANSWER

Conjugate, then transpose

$$\langle\psi| = \frac{1}{5}\begin{bmatrix}3 & -4i\end{bmatrix}$$

$\left|\frac{3}{5}\right|^2 + \left|\frac{4i}{5}\right|^2 = \frac{9}{25} + \frac{16}{25} = 1$: normalised.

The i becomes −i in the bra.

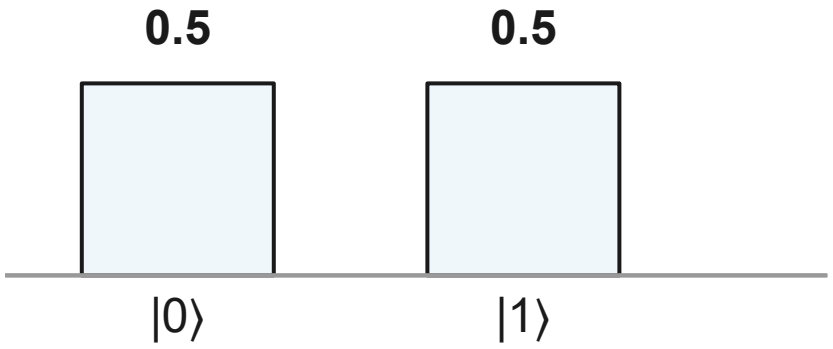
LIVE DEMO

One state, three tools

```
amps = np.array([1, 1j]) / np.sqrt(2)
sv = Statevector(amps)
psi = qt.Qobj(amps.reshape(2, 1))
p_qk = sv.probabilities()
p_qt = np.abs(psi.full().ravel()) ** 2
```

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

Install the environment from handout section 1.10 before Lecture 2.




```
NumPy  probs: [0.5 0.5]
Qiskit probs: [0.5 0.5]
QuTiP  probs: [0.5 0.5]
same amplitudes: True
```

Recap

1

The memory wall

n qubits need 2^n amplitudes: 16 GiB at 30.

2

Amplitudes can cancel

Probabilities only add; complex amplitudes interfere.

3

Phasors → amplitudes

$z^*z = |z|^2$; i replaces j.

4

Kets and bras

$\langle\psi|$ = conjugate row; normalise with $|\alpha|^2 + |\beta|^2$

Next: Lecture 2

Inner products, operators and eigenvalues

Read: handout sections 1.1–1.4; install the simulators (section 1.10).

Exit ticket

Normalise $(1 - i)|0\rangle + |1\rangle$ and write its bra.

Copyright © 2026 Sajid Muhaimin Choudhury. All rights reserved.

Rights and attributions

Copyright (c) 2026 Sajid Muhaimin Choudhury. All rights reserved.

Public edition of Lecture 1: Why Quantum Computing?. Release qc-2026-10-15, prepared 2026-10-15 from source revision 3df9449aa6.

This edition omits speaker notes, instructor-only material and every figure whose publication rights are not cleared for public distribution.