Fermilab | U.S. DEPARTMENT OF ENERGY | Office of Science

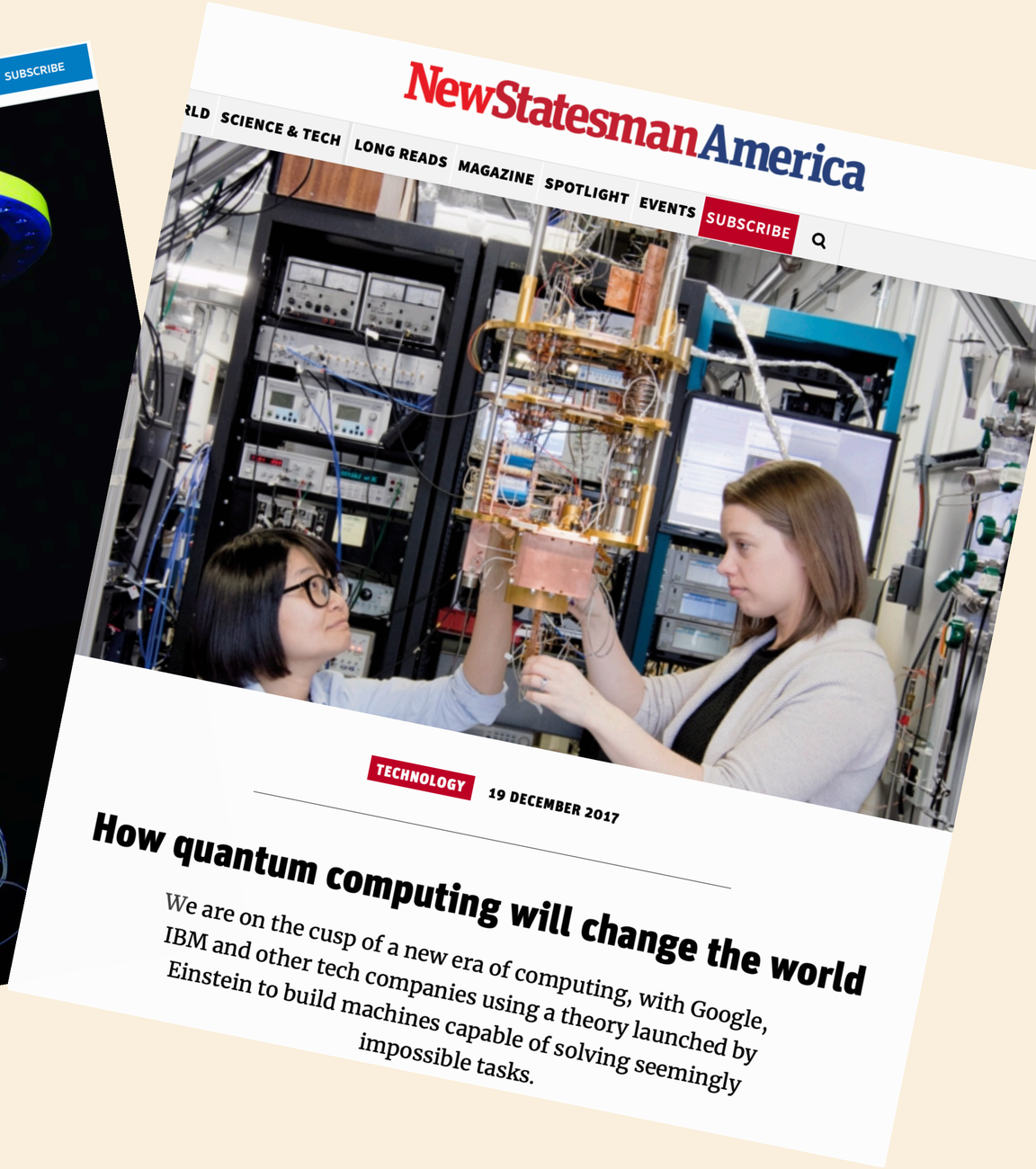
Introduction to Quantum Computing

Adam Lyon

2019 Undergraduate Summer Lecture Series

8 August 2019

Hype



Fermilab

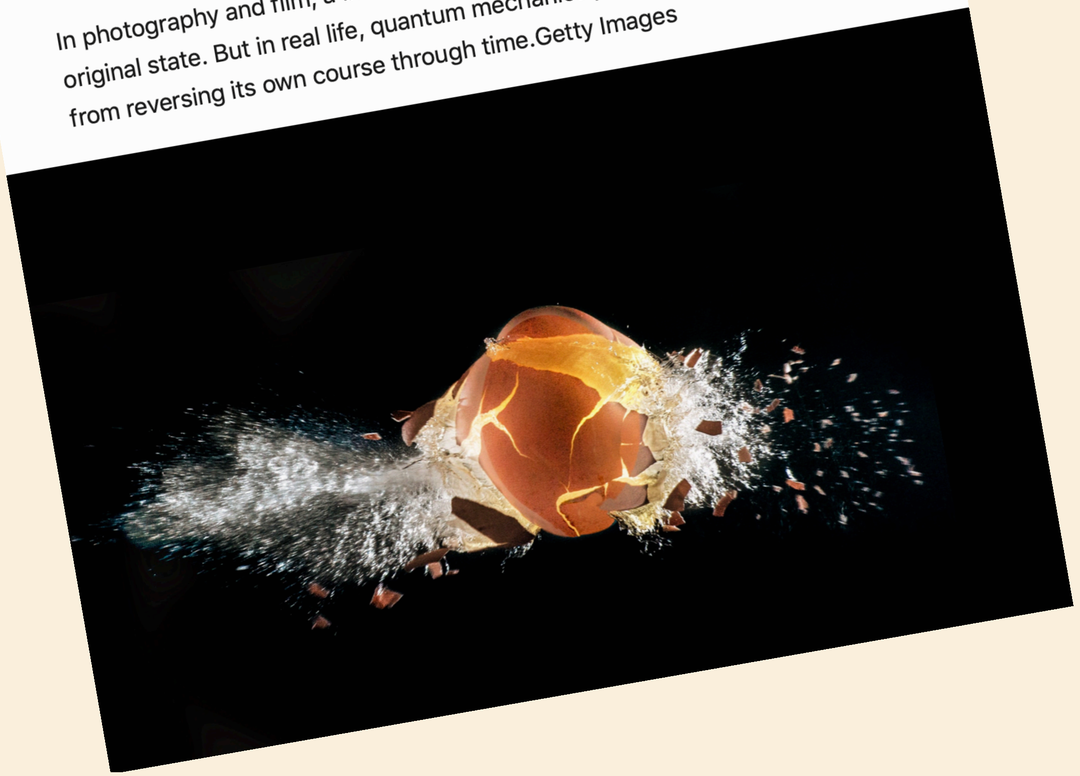
Hyper hype

NYT

For a Split Second, a Quantum Computer Made History Go Backward

By Dennis Overbye • May 8, 2019

In photography and film, a broken egg can be perfectly unscrambled to its original state. But in real life, quantum mechanics prevent even a single particle from reversing its own course through time.Getty Images

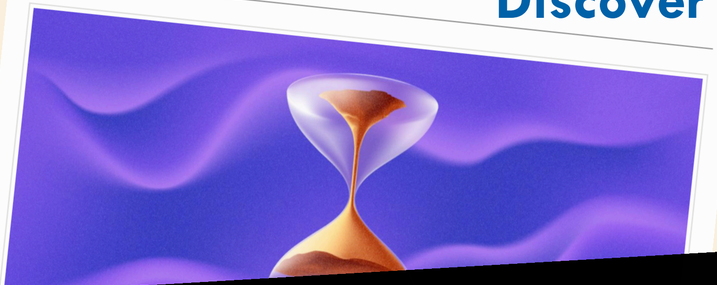


Scientists Used IBM's Quantum Computer to Reverse Time, Possibly Breaking a Law of Physics

By Korey Haynes | March 13, 2019 11:28 am

947

Discover



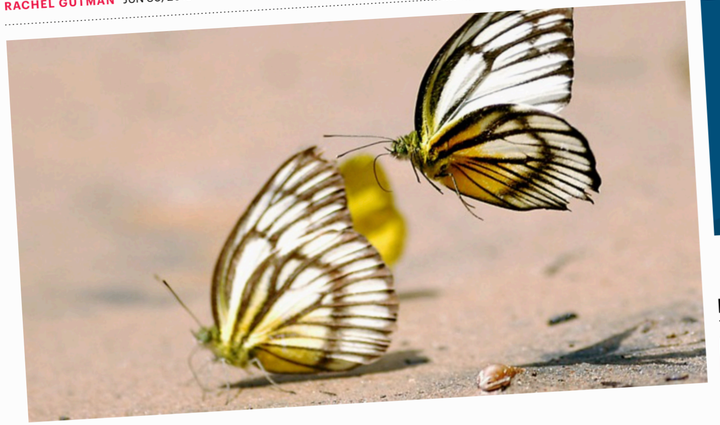
The Atlantic Popular Latest Sections Magazine Subscribe

TECHNOLOGY

Could Quantum Computing Be the End of Free Will?

On the fear that too much processing power will make us cease to be human

RACHEL GUTMAN JUN 30, 2018



YUSUF AHMAD / REUTERS

MORE IN THIS SERIES

Fermilab

Super-hyper hype

IEEE SPECTRUM Topics Reports Blogs Multimedia

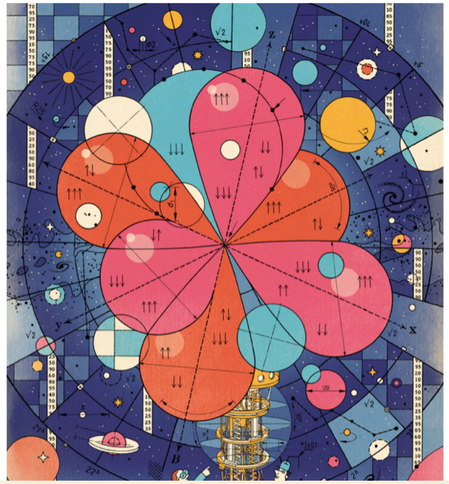
Feature | Computing | Hardware

15 Nov 2018 | 16:00 GMT

The Case Against Quantum Computing

The proposed strategy relies on manipulating with high precision an unimaginably huge number of variables

By **Mikhail Dyakonov**



Quantum computing is all the rage. It seems like hardly a day goes by without some news outlet describing the extraordinary things this technology promises. Most commentators forget, or just gloss over, the fact that people have been working on quantum computing for decades—and without any practical results to show for it.

We've been told that quantum computers could "provide breakthroughs in many disciplines

GIZMODO We come from the future

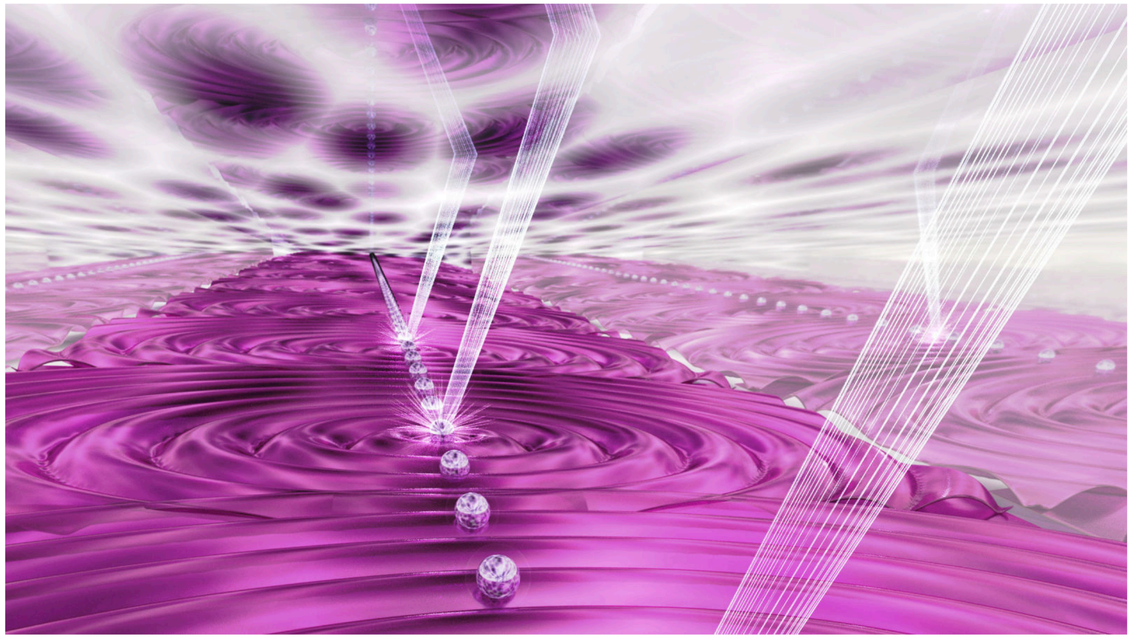
LATEST SCIENCE REVIEWS IO9 FIELD GUIDE EARTHER DESIGN PALEOFUTURE

PHYSICS

It's Time to Plan for How Quantum Computing Could Go Wrong, Say Entrepreneurs and Physicists

Ryan F. Mandelbaum
12/13/18 6:00pm • Filed to: ETHICS

29.1K 19 4



More of that weird quantum art that always comes with quantum computing press releases.
Illustration: E. Edwards/JQI (NIST)

Fermilab

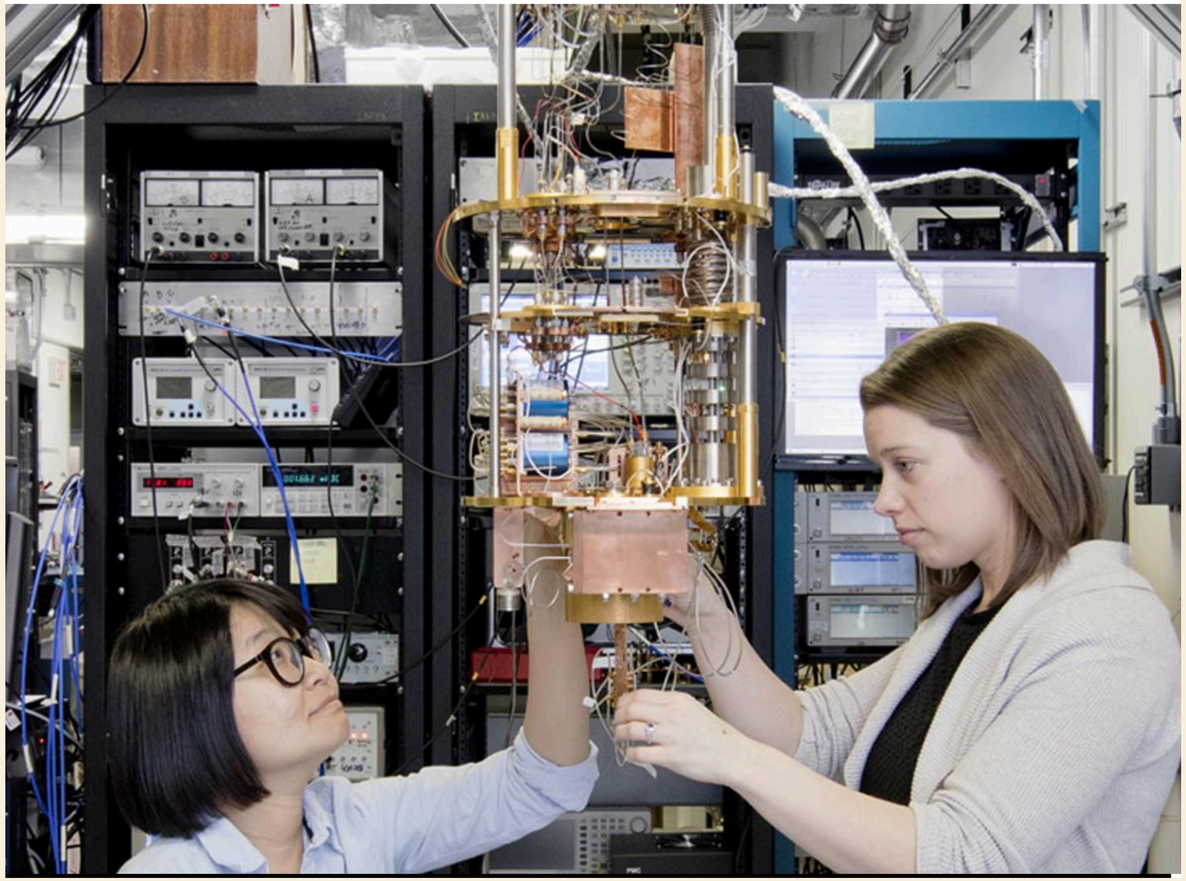
By the way (seems to be an effort to highlight women in QC)

IBM Q

Quantum starts here

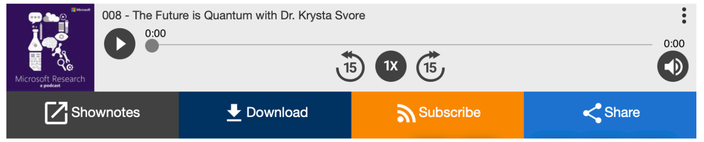
IBM Q is the world's most advanced quantum computing initiative, focused on propelling the science and pioneering commercial applications for quantum advantage.

Learn more about quantum computing at IBM →



The future is quantum with Dr. Krysta Svore

January 17, 2018 | By Microsoft blog editor



Microsoft Principal Research Manager Dr. Krysta Svore. Photography by Maryatt Photography.

Fermilab

The real question I'll try to answer

Fermilab

Overview

- Motivate the need for a new kind of computing…
  - Learn about computational complexity
  - Traditional (Classical) computers are good at certain kinds of problems and bad at others
- Learn just enough quantum mechanics

  "I think I can safely say that nobody understands quantum mechanics" — Richard Feynman
- Explain the building blocks of Quantum Computers – Qubits
- Learn the very basics of how Quantum Computers work
- Discuss applications and where things are today

A little about me

17 years as a Scientist at Fermilab
Undergrad at N.C. State University
Graduate School at U of Maryland & Fermilab
Postdoc at U of Rochester @ Cornell

VOLUME 83, NUMBER 24 PHYSICAL REVIEW LETTERS 13 DECEMBER 1999

Search for Squarks and Gluinos in Events Containing Jets and a Large Imbalance in Transverse Energy

B. Abbott,[40] M. Abolins,[37] V. Abramov,[15] B. S. Acharya,[8] I. Adam,[39] D. L. Adams,[49] M. Adams,[24] S. Ahn,[23] G. A. Alves,[2] N. Amos,[36] E. W. Anderson,[30] M. M. Baarmand,[42] V. V. Babintsev,[15] L. Babukhadia,[16] A. Baden,[33] B. Baldin,[23] S. Banerjee,[8] J. Bantly,[46] E. Barberis,[17] P. Baringer,[31] J. F. Bartlett,[23] A. Belyaev,[14] S. B. Beri,[6] I. Bertram,[26] V. A. Bezzubov,[15] P. C. Bhat,[23] V. Bhatnagar,[6] M. Bhattacharjee,[42] N. Biswas,[28] G. Blazey,[25] S. Blessing,[21] P. Bloom,[18] A. Boehnlein,[23] N. I. Bojko,[15] F. Borcherding,[23] C. Boswell,[20] A. Brandt,[23] R. Breedon,[18] G. Briskin,[46] R. Brock,[37] A. Bross,[23] D. Buchholz,[26] V. S. Burtovoi,[15] J. M. Butler,[34] W. Carvalho,[2] D. Casey,[37] Z. Casilum,[42] H. Castilla-Valdez,[11] D. Chakraborty,[42] S. V. Chekulaev,[15] W. Chen,[42] S. Choi,[10] S. Chopra,[21] B. C. Choudhary,[20] J. H. Christenson,[23] M. Chung,[24] D. Claes,[38] A. R. Clark,[17] W. G. Cobau,[33] J. Cochran,[20] L. Coney,[28] W. E. Cooper,[23] D. Coppage,[31] C. Cretsinger,[41] D. Cullen-Vidal,[46] M. A. C. Cummings,[25] D. Cutts,[46] O. I. Dahl,[17] K. Davis,[16] K. De,[47] K. Del Signore,[36] M. Demarteau,[23] D. Denisov,[23] S. P. Denisov,[15] H. T. Diehl,[23] M. Diesburg,[23] G. Di Loreto,[37] P. Draper,[47] Y. Ducros,[5] L. V. Dudko,[14] S. R. Dugad,[8] A. Dyshkant,[15] D. Edmunds,[37] J. Ellison,[20] V. D. Elvira,[42] R. Engelmann,[42] S. Eno,[33] G. Eppley,[49] P. Ermolov,[14] O. V. Eroshin,[15] V. N. Evdokimov,[15] T. Fahland,[19] M. K. Fatyga,[41] S. Feher,[23] D. Fein,[16] T. Ferbel,[41] H. E. Fisk,[23] Y. Fisyak,[43] E. Flattum,[23] G. E. Forden,[16] M. Fortner,[25] K. C. Frame,[37] S. Fuess,[23] E. Gallas,[47] A. N. Galyaev,[15] P. Gartung,[20] V. Gavrilov,[13] T. L. Geld,[37] R. J. Genik II,[37] K. Genser,[23] C. E. Gerber,[23] Y. Gershtein,[13] B. Gibbard,[43] B. Gobbi,[26] B. Gómez,[4] G. Gómez,[33] P. I. Goncharov,[15] J. L. González Solís,[11] H. Gordon,[43] L. T. Goss,[48] K. Gounder,[20] A. Goussiou,[42] N. Graf,[43] P. D. Grannis,[42] D. R. Green,[23] H. Greenlee,[23] S. Grinstein,[1] P. Grudberg,[17] S. Grünendahl,[23] G. Guglielmo,[45] J. A. Guida,[16] J. M. Guida,[46] A. Gupta,[8] S. N. Gurzhiev,[15] G. Gutierrez,[23] P. Gutierrez,[45] N. J. Hadley,[33] H. Haggerty,[23] S. Hagopian,[21] V. Hagopian,[21] K. S. Hahn,[41] R. E. Hall,[19] P. Hanlet,[35] S. Hansen,[23] J. M. Hauptman,[30] C. Hebert,[31] D. Hedin,[25] A. P. Heinson,[20] U. Heintz,[34] R. Hernández-Montoya,[11] T. Heuring,[21] R. Hirosky,[24] J. D. Hobbs,[42] B. Hoeneisen,[4,*] J. S. Hoftun,[46] F. Hsieh,[36] Tong Hu,[27] A. S. Ito,[23] J. Jaques,[28] S. A. Jerger,[37] R. Jesik,[27] T. Joffe-Minor,[26] K. Johns,[16] M. Johnson,[23] A. Jonckheere,[23] M. Jones,[22] H. Jöstlein,[23] S. Y. Jun,[26] C. K. Jung,[42] S. Kahn,[43] G. Kalbfleisch,[45] D. Karmanov,[14] D. Karmgard,[21] R. Kehoe,[28] S. K. Kim,[10] B. Klima,[23] C. Klopfenstein,[18] W. Ko,[18] J. M. Kohli,[6] D. Koltick,[29] A. V. Kostritskiy,[15] J. Kotcher,[43] A. V. Kotwal,[39] A. V. Kozelov,[15] E. A. Kozlovsky,[15] J. Krane,[38] M. R. Krishnaswamy,[8] S. Krzywdzinski,[23] S. Kuleshov,[13] Y. Kulik,[42] S. Kunori,[33] F. Landry,[37] G. Landsberg,[46] B. Lauer,[30] A. Leflat,[14] J. Li,[47] Q. Z. Li,[23] J. G. R. Lima,[3] D. Lincoln,[23] S. L. Linn,[21] J. Linnemann,[37] R. Lipton,[23] F. Lobkowicz,[41] A. Lucotte,[42] L. Lueking,[23] A. L. Lyon,[33] A. K. A. Maciel,[2] R. J. Madaras,[17] R. Madden,[21] L. Magaña-Mendoza,[11] V. Manankov,[14] S. Mani,[18] H. S. Mao,[23,†] R. Markeloff,[25] T. Marshall,[27] M. I. Martin,[23] K. M. Mauritz,[30] B. May,[26] A. A. Mayorov,[15] R. McCarthy,[42] J. McDonald,[21] T. McKibben,[24] J. McKinley,[37] T. McMahon,[44] H. L. Melanson,[23] M. Merkin,[14] K. W. Merritt,[23] C. Miao,[46] H. Miettinen,[49] A. Mincer,[40] C. S. Mishra,[23] N. Mokhov,[23] N. K. Mondal,[8] H. E. Montgomery,[23] P. Mooney,[4] M. Mostafa,[1] H. da Motta,[2] C. Murphy,[24] F. Nang,[16] M. Narain,[34] V. S. Narasimham,[8] A. Narayanan,[16] H. A. Neal,[36] J. P. Negret,[4] P. Nemethy,[40] D. Norman,[48] L. Oesch,[36] V. Oguri,[3] N. Oshima,[23] D. Owen,[37] P. Padley,[49] A. Para,[23] N. Parashar,[35] Y. M. Park,[9] R. Partridge,[46] N. Parua,[8] M. Paterno,[41] B. Pawlik,[12] J. Perkins,[47] M. Peters,[22] R. Piegaia,[1] H. Piekarz,[21] Y. Pischalnikov,[29] B. G. Pope,[37] H. B. Prosper,[21] S. Protopopescu,[43] J. Qian,[36] P. Z. Quintas,[23] R. Raja,[23] S. Rajagopalan,[43] O. Ramirez,[24] S. Reucroft,[35] M. Rijssenbeek,[42] T. Rockwell,[37] M. Roco,[23] P. Rubinov,[26] R. Ruchti,[28] J. Rutherfoord,[16] A. Sánchez-Hernández,[11] A. Santoro,[2] L. Sawyer,[32] R. D. Schamberger,[42] H. Schellman,[26] J. Sculli,[40] E. Shabalina,[14] C. Shaffer,[21] H. C. Shankar,[8] R. K. Shivpuri,[7] D. Shpakov,[42] M. Shupe,[16] H. Singh,[20] J. B. Singh,[6] V. Sirotenko,[25] E. Smith,[45] R. P. Smith,[23] R. Snihur,[26] G. R. Snow,[38] J. Snow,[44] S. Snyder,[43] J. Solomon,[24] M. Sosebee,[47] N. Sotnikova,[14] M. Souza,[2] G. Steinbrück,[45] R. W. Stephens,[47] M. L. Stevenson,[17] F. Stichelbaut,[43] D. Stoker,[19] V. Stolin,[13] D. A. Stoyanova,[15] M. Strauss,[45] K. Streets,[40] M. Strovink,[17] A. Sznajder,[2] P. Tamburello,[33] J. Tarazi,[19] M. Tartaglia,[23] T. L. T. Thomas,[26] J. Thompson,[33] T. G. Trippe,[17] P. M. Tuts,[39] V. Vaniev,[15] N. Varelas,[24] E. W. Varnes,[17] A. A. Volkov,[15] A. P. Vorobiev,[15] H. D. Wahl,[21] G. Wang,[21] J. Warchol,[28] G. Watts,[46] M. Wayne,[28] H. Weerts,[37] A. White,[47] J. T. White,[48] J. A. Wightman,[30] S. Willis,[25] S. J. Wimpenny,[20] J. V. D. Wirjawan,[48] J. Womersley,[23] E. Won,[41] D. R. Wood,[35] Z. Wu,[23,†] R. Yamada,[23] P. Yamin,[43] T. Yasuda,[35] P. Yepes,[49] K. Yip,[23] C. Yoshikawa,[22] S. Youssef,[21] J. Yu,[23] Y. Yu,[10] B. Zhang,[23,†] Z. Zhou,[30] Z. H. Zhu,[41] M. Zielinski,[41] D. Zieminska,[27] A. Zieminski,[27] E. G. Zverev,[14] and A. Zylberstejn[5]

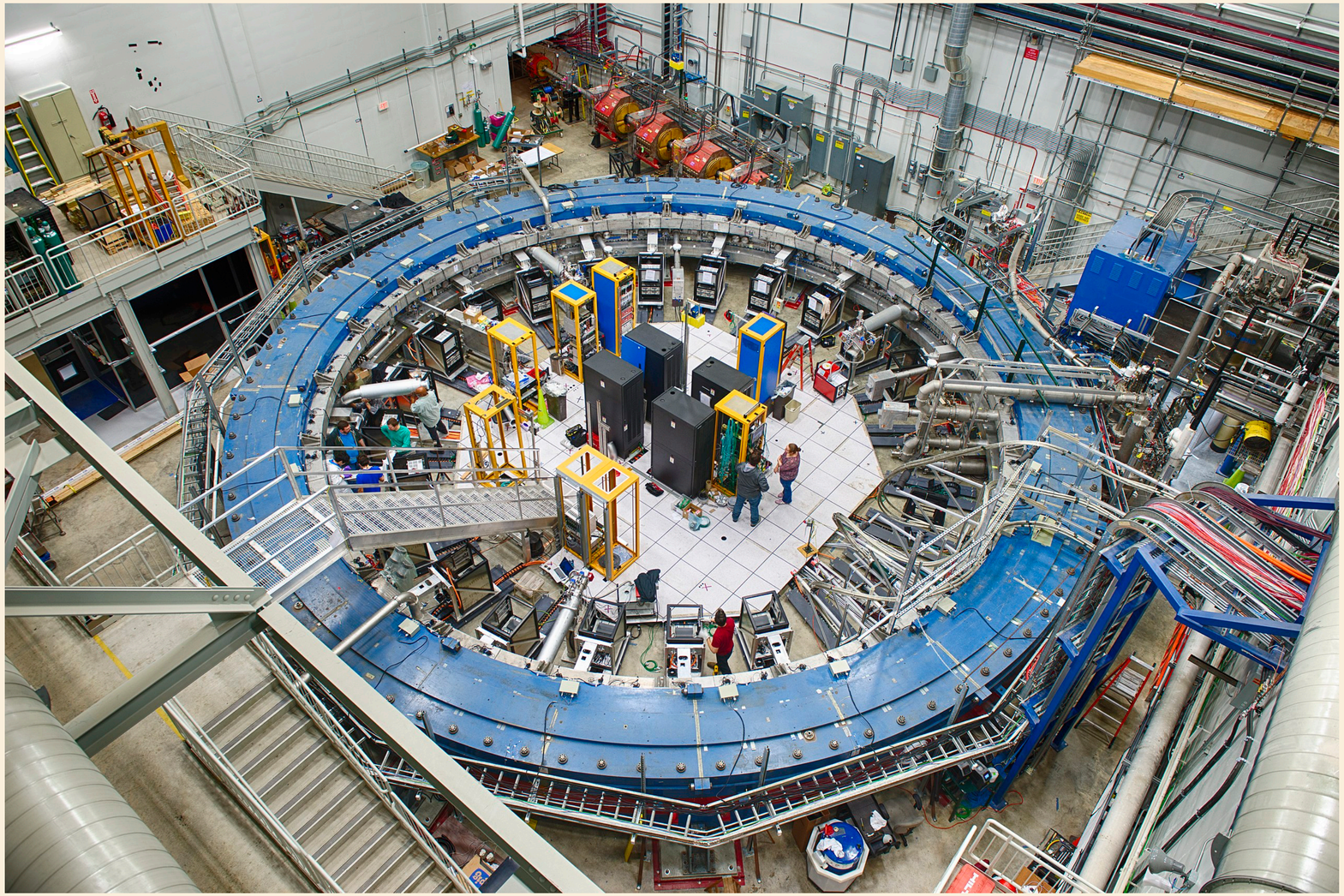
(D0 Collaboration)

[1]*Universidad de Buenos Aires, Buenos Aires, Argentina*
[2]*LAFEX, Centro Brasileiro de Pesquisas Físicas, Rio de Janiero, Brazil*
[3]*Universidade do Estado do Rio de Janeiro, Rio de Janeiro, Brazil*

0031-9007/99/83(24)/4937(6)$15.00

What do I do?

- Scientific Computing (management)
- Muon g-2 Experiment
- Getting into Quantum Computing
- Enough about me…

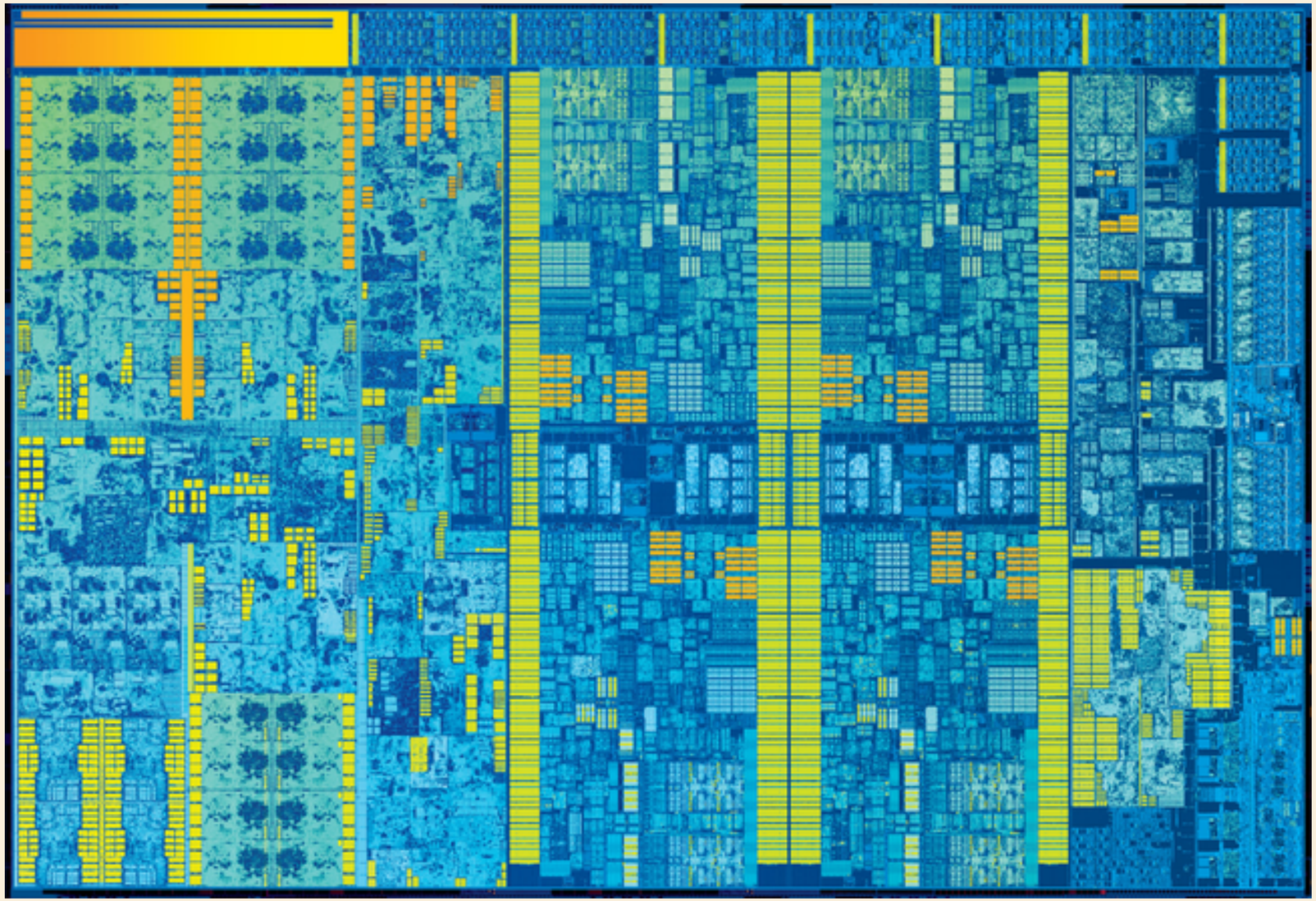


Fermilab

Note: All computers today need Quantum (QM)

- Intel Skylake Processor
- 28 cores
- 8 BILLION transistors
- Quantum mechanics governs construction and limitations
- The materials science is based on QM

- BUT – operation is not “quantum”
- Manipulate bits: (0 or 1)
- Classical Digital Machine

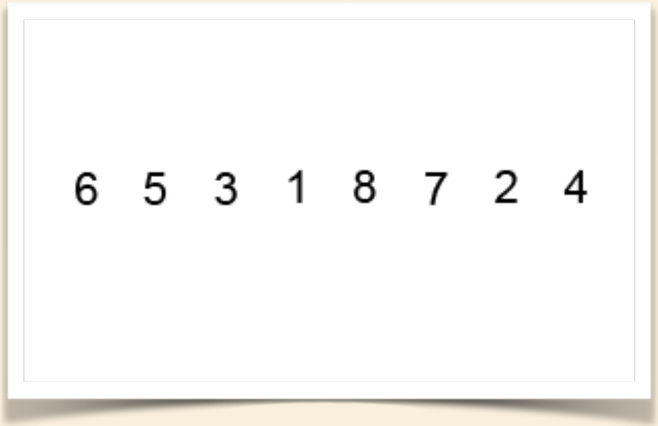


Computers run algorithms

- **A process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer.** Google

- **e.g. Bubble Sort** https://github.com/gibsjose/cpp-cheat-sheet

```
swapped = true
while swapped
  swapped = false
  for j from 0 to N - 1
     if a[j] > a[j + 1]
        swap( a[j], a[j + 1] )
        swapped = true
```

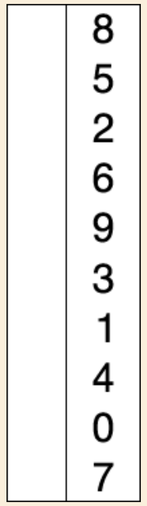


Fermilab

How do we compare algorithms? Complexity

- Insertion

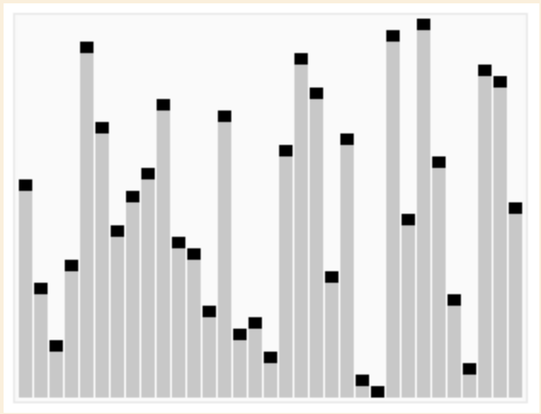
- Selection



- Bubble

- Merged

- QuickSort



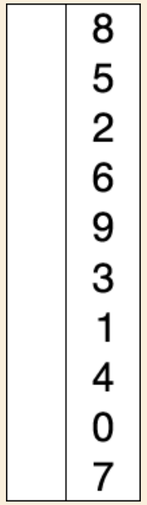
https://github.com/gibsjose/cpp-cheat-sheet

Fermilab

How do we compare algorithms? Complexity (asymptotic)

- **Insertion**
  $O(n^2)$

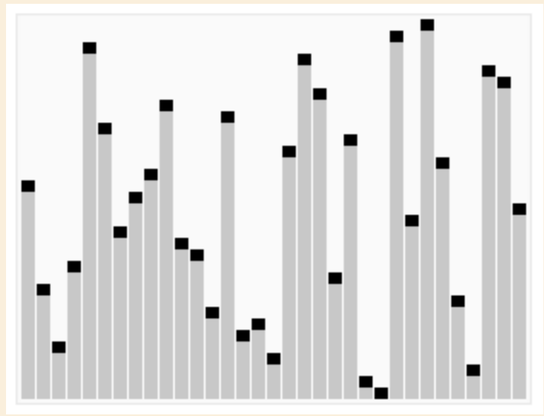
- **Selection**
  $O(n^2)$



- **Bubble**
  $O(n^2)$

- **Merged**
  $O(n \log n)$

- **QuickSort**
  $O(n \log n)$



Asymptotic number of operations

https://github.com/gibsjose/cpp-cheat-sheet

Fermilab

Complexity (C++ Standard Template Library)

Data Structures

Data Structure	Time Complexity								Space Complexity
	Average				Worst				Worst
	Indexing	Search	Insertion	Deletion	Indexing	Search	Insertion	Deletion	
Basic Array	O(1)	O(n)	-	-	O(1)	O(n)	-	-	O(n)
Dynamic Array	O(1)	O(n)	O(n)	O(n)	O(1)	O(n)	O(n)	O(n)	O(n)
Singly-Linked List	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)	O(n)
Doubly-Linked List	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)	O(n)
Skip List	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	O(n)	O(n)	O(n)	O(n log(n))
Hash Table	-	O(1)	O(1)	O(1)	-	O(n)	O(n)	O(n)	O(n)
Binary Search Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	O(n)	O(n)	O(n)	O(n)
Cartresian Tree	-	O(log(n))	O(log(n))	O(log(n))	-	O(n)	O(n)	O(n)	O(n)
B-Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)
Red-Black Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)
Splay Tree	-	O(log(n))	O(log(n))	O(log(n))	-	O(log(n))	O(log(n))	O(log(n))	O(n)
AVL Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)

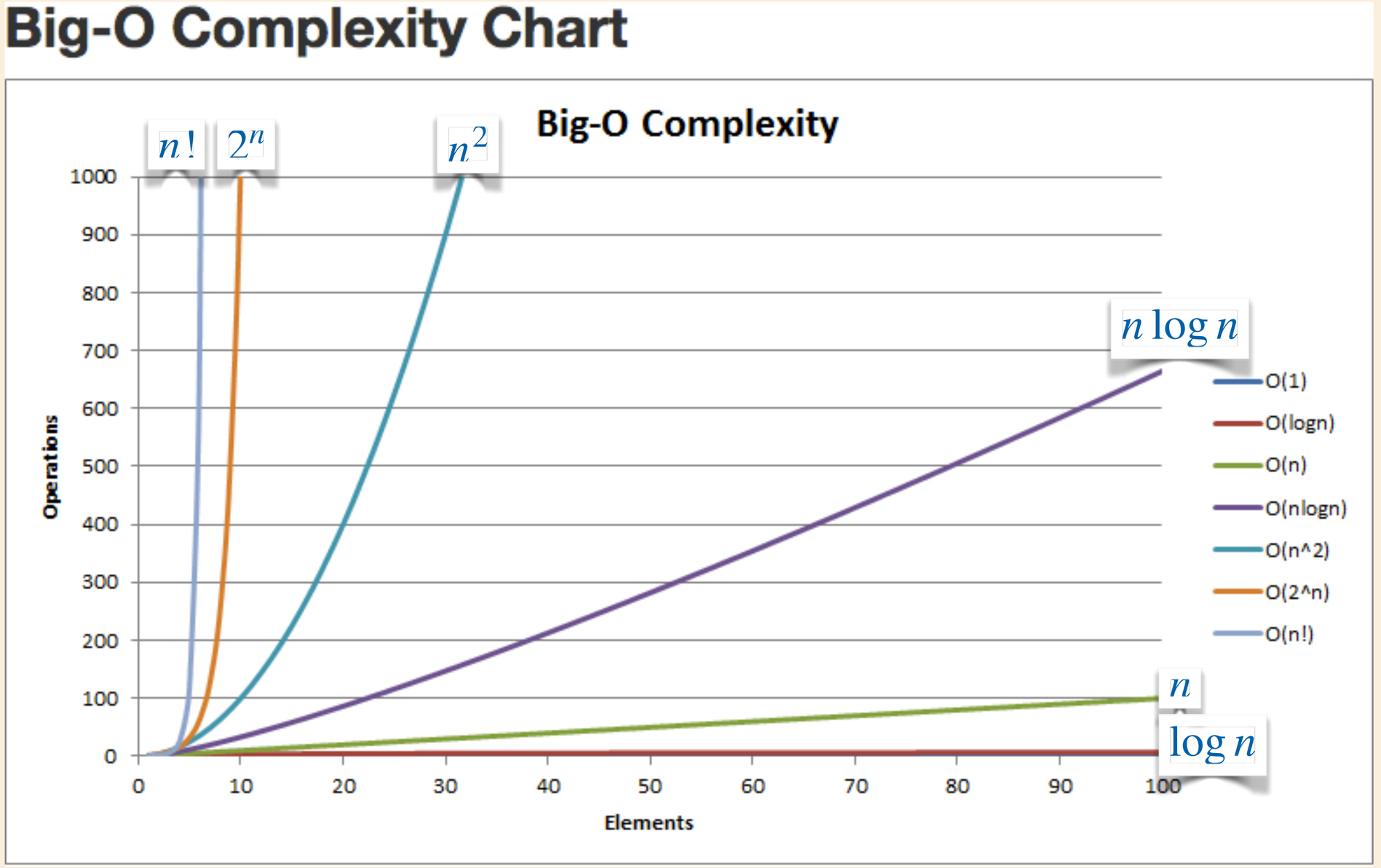
https://github.com/gibsjose/cpp-cheat-sheet

Fermilab

Complexity

$O(1), O(\log n), O(n), O(n \log n), O(n^2)$ **scale as polynomial, even** $O(n^{1,000,000})$

2^n **scales as exponential**



https://github.com/gibsjose/cpp-cheat-sheet

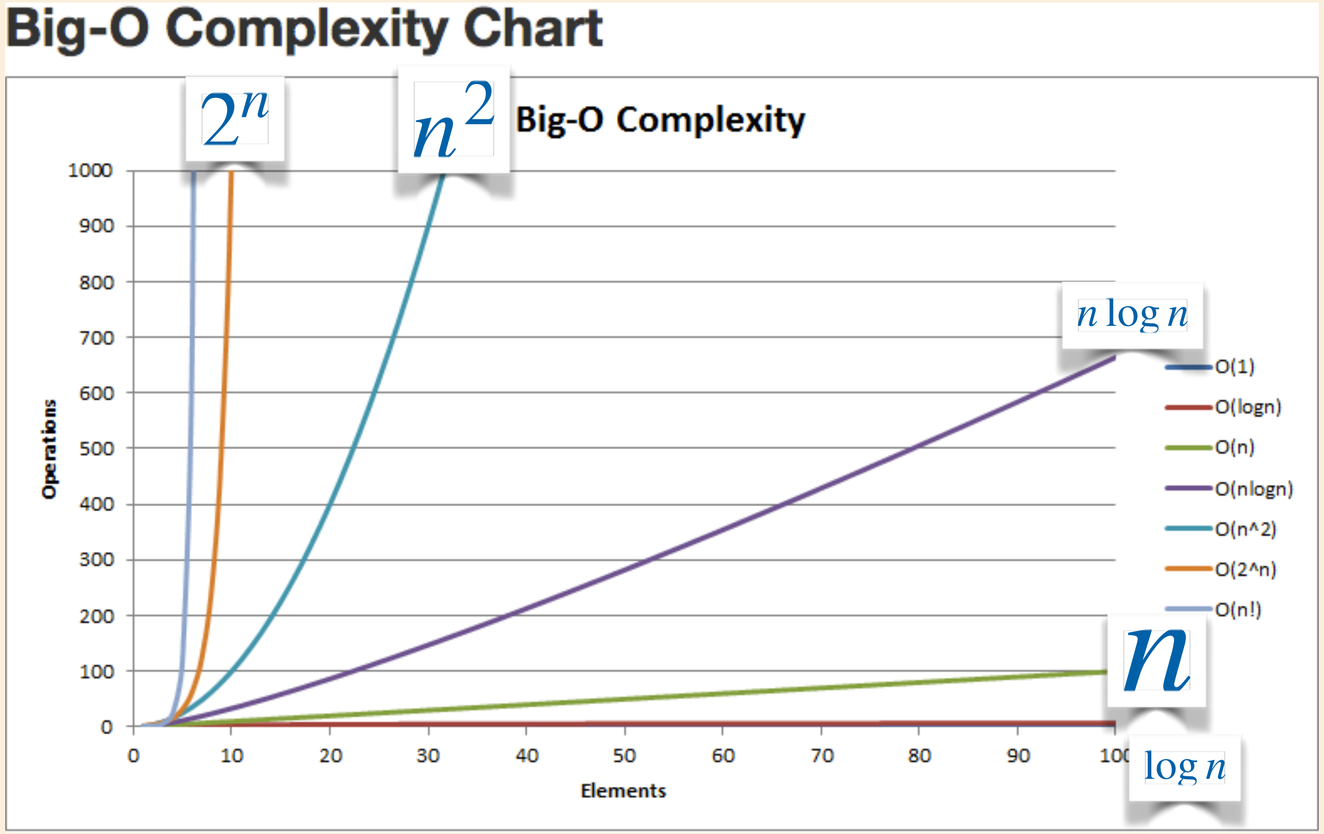
Fermilab

Complexity of Problems

P (Polynomial)

Solvable with number of steps scaling at most like size raised to a *fixed* power, like $O(n^2)$

All problems currently **solvable** by traditional computers are *P (efficient)*



https://github.com/gibsjose/cpp-cheat-sheet

NP (Nondeterministic Polynomial) e.g. $O(2^n)$

May not be solvable by an efficient algorithm but an answer can be checked efficiently.
…Can't find the solution with *P*, but can verify with *P*

Current traditional computers, even the fastest future supercomputers, **cannot solve large** *NP* problems in a reasonable amount of time

Fermilab

An NP problem - factorizing numbers

$15 = 5 \times 3$ $91 = 13 \times 7$ $4095 = 45 \times 91 = 13 \times 7 \times 5 \times 3 \times 3$

RSA Encryption (Rivest, Shamir, Adleman)

You want send your credit card information to a web-store securely

The store sends you a *public key* (a big number) that the algorithm uses to encrypt your message.

The store uses the corresponding top secret *private key* they hold to decrypt your message.

The public key is not usable for decrypting messages, only encrypting

How does this work? (Rivest, Shamir, Adleman)

The private key is two large prime numbers.

The public key is their *product*

So long as the public key is hard to factor, anyone can encrypt but only the store can decrypt

`https://` uses this technique. So long as the public key is big, you can't figure out the private key

RSA Encryption and Factorization

- Factorization is an NP problem (very hard to solve, but efficient to check)
- Best algorithm is "General Number Field Sieve" $O\left(e^{\sqrt[3]{b(\log b)^2}}\right)$, basically like $O(2^n)$
- Largest integer factorized is RSA-768 (768 bits, 232 decimal digits) in 2009
  - 100s of machines running for over two years - equivalent to 2000 CPU years
- RSA-2048 is the current standard (617 decimal digits)

~6 quadrillion years to break on a desktop
~6 billion years on a million processor supercomputer
Expected to remain unbroken until after 2030
Then we'll use RSA-4096 !
Note that RSA-1024 was expected to be broken before 2010 and it hasn't yet

```
RSA-768 = 123018668453011775513049495838496272077285356959533479219732245215172640050726365751874520219978646938995647494277406384592519255732630345373154826850791702612214291346167042921431160222124047927473779408066535141959745985690214341 3
```

```
RSA-768 = 33478071698956898786044169848212690817704794983713768568912431388982883793878002287614711652531743087737814467999489
        × 36746043666799590428244633799627952632279158164343087642676032283815739666511279233373417143396810270092798736308917
```

https://www.digicert.com/TimeTravel/math.htm
https://en.wikipedia.org/wiki/Integer_factorization
https://en.wikipedia.org/wiki/RSA_numbers#RSA-768

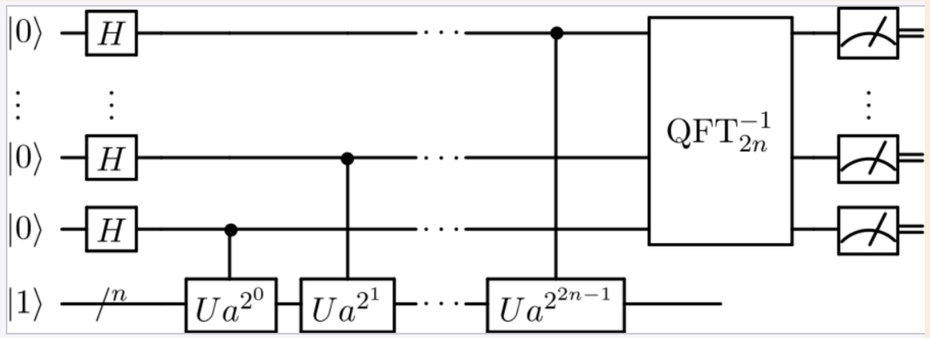
Fermilab

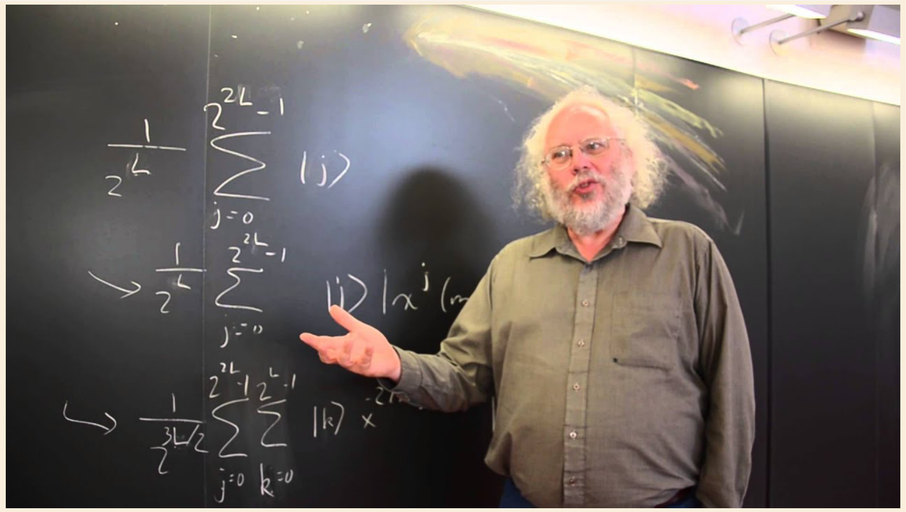
And now for something completely different

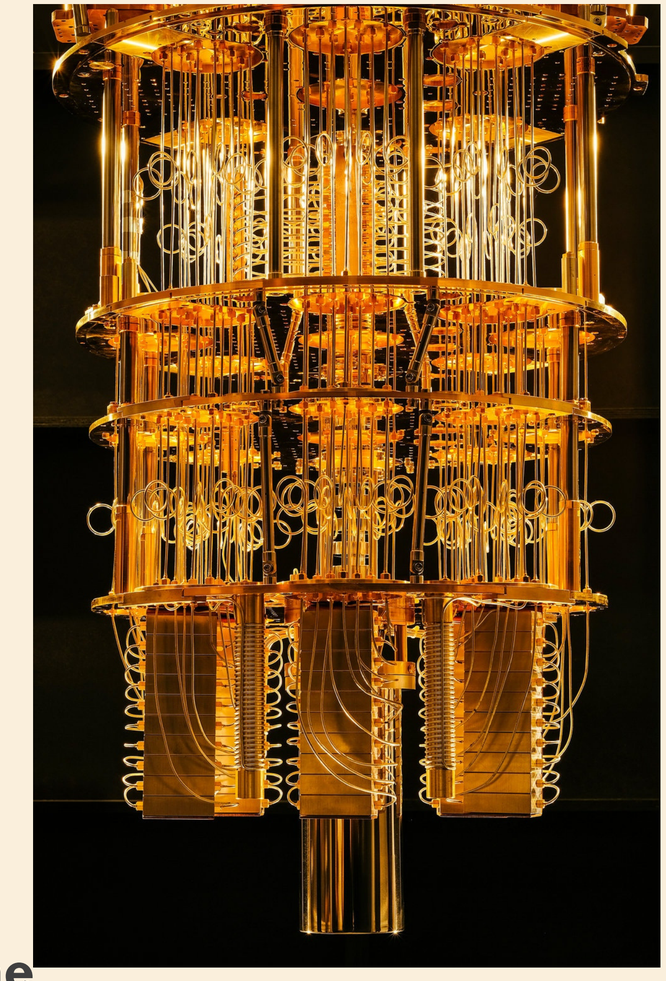
Think about a completely different type of computer… a Quantum Computer

Shor's Algorithm invented in 1994 by Peter Shor (MIT) can factor numbers on a Quantum Computer with polynomial scaling

$$O(b^2 \log b \log\log b) \sim O(b^3)$$







The largest number factored with Shor's algorithm is 21 = 7 x 3 (wow!)

Largest number factored by a later algorithm is 56,153 = 241 x 233
Could potentially factor very large numbers in a short amount of time

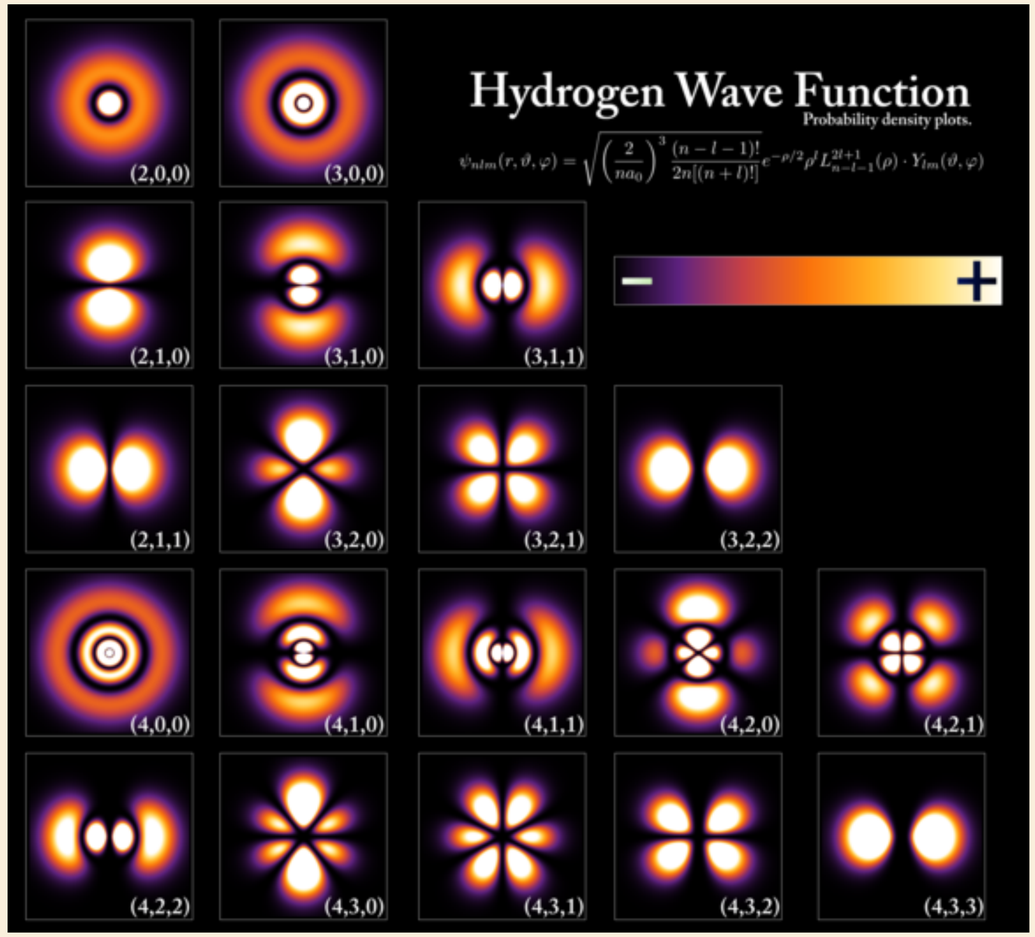
Fermilab

Quantum Computing

Potential for Quantum Computing to ruin RSA, but not any time soon.

This isn't the thing that drives Quantum Computing now, but allows us to introduce it…

A Quantum Computer OPERATES exploiting the strange world of Quantum Mechanics



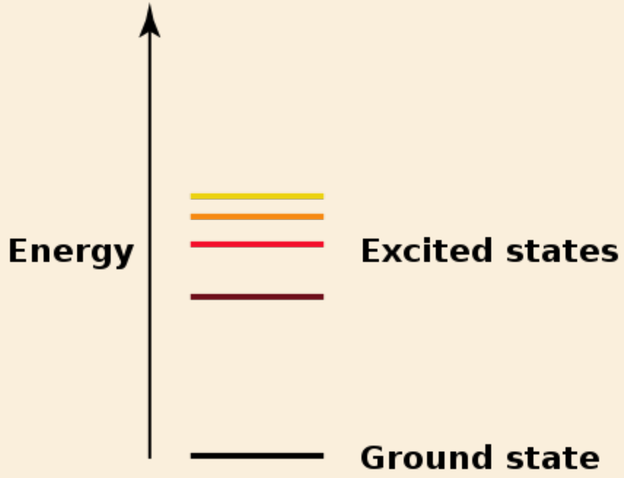
http://bit.ly/2JWA1GM

Quick Introduction to just enough Quantum Mechanics
(QC Algorithms are very involved - so we'll learn the basics)

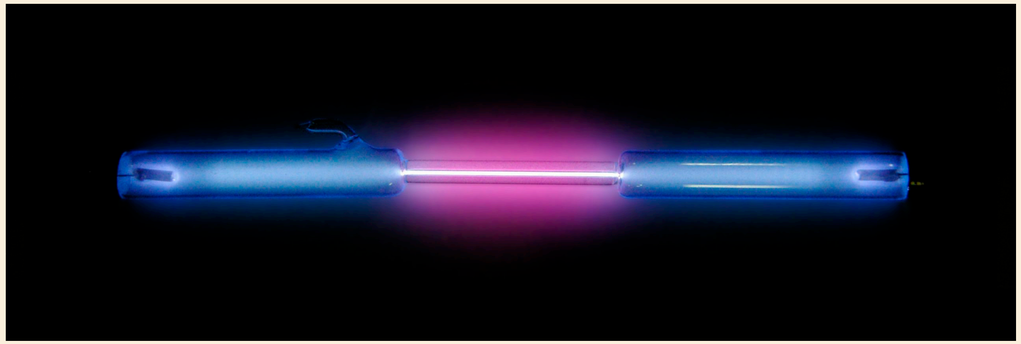
Fermilab

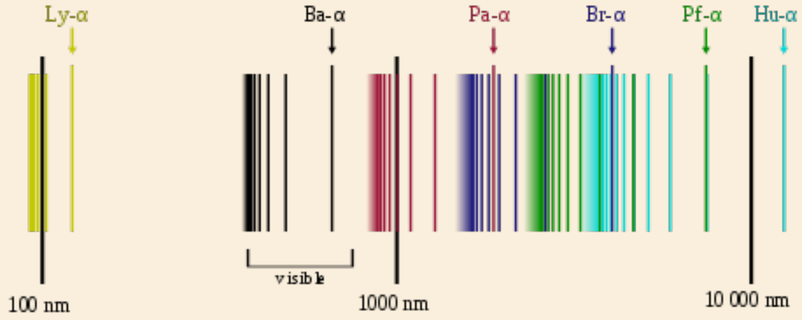
Important concepts of Quantum Mechanics

- **States - A system is in a state with an energy. The ground state is the state with the lowest energy. Changing states requires absorbing or emitting a *specific* amount of energy.**

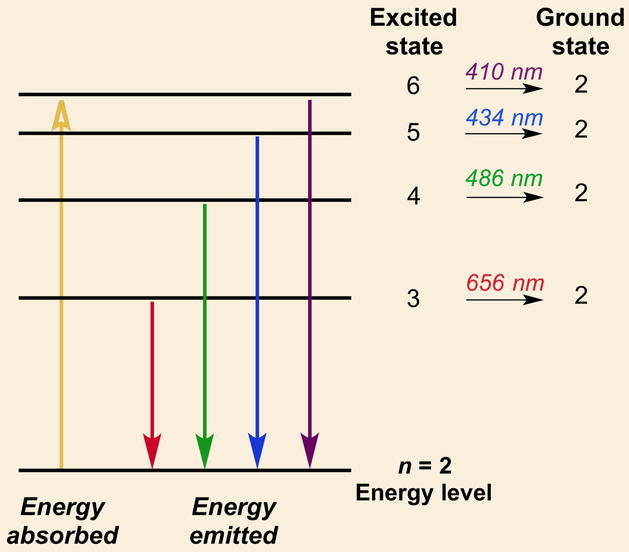


- **For example, electron levels in a Hydrogen atom**





http://bit.ly/2YpBQVV



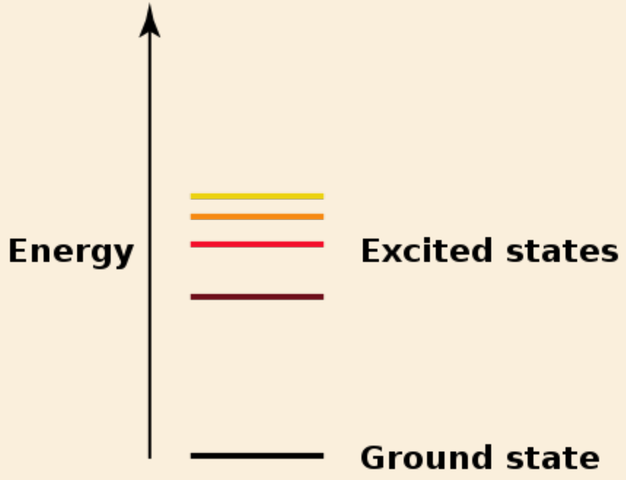
http://bit.ly/2YkKyoh

http://bit.ly/2JZzgwx

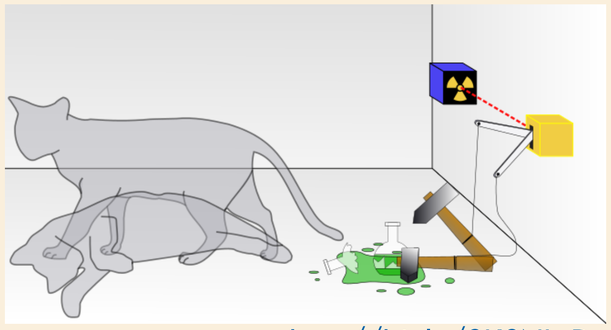
Fermilab

Important concepts of Quantum Mechanics

- States - A system is in a state with an energy. The ground state is the state with the lowest energy. Changing states requires absorbing or emitting a *specific* amount of energy.



- Superposition of states - A system can be in more than one state AT THE SAME TIME [strange!] Each possible state has an associated *amplitude*



http://bit.ly/2K2NLzD

- The sum of all the possible states is the *wave function* $|\psi\rangle = \alpha\,|\text{dead}\rangle + \beta\,|\text{alive}\rangle$

- Upon measurement, you will observe ONE state. QM can tell us its probability e.g. Open the box and you'll see the cat alive with probability β^2

Fermilab

Quantum Mechanics

- **The probability of *observing* a system in a particular state is that state's $|\text{amplitude}|^2$**

- **Note: Probabilities follow certain rules…**

  $P(\text{rain}) \geq 0, P(\text{no rain}) \geq 0$ **Negative makes no sense**

  $P(\text{rain}) + P(\text{no rain}) = 1$ **Something must happen**

- **The only rule amplitudes must follow is**

  $|\alpha|^2 + |\beta|^2 = 1$ **Something must happen**

Miami, FL 5 Day Weather

11:30 pm EDT Print

DAY	DESCRIPTION	HIGH / LOW	PRECIP
TONIGHT JUL 23	Partly Cloudy	--/78°	20%
WED JUL 24	PM Thunderstorms	91°/78°	50%
THU JUL 25	PM Thunderstorms	89°/78°	80%
FRI JUL 26	PM Thunderstorms	89°/80°	40%
SAT JUL 27	Scattered Thunderstorms	88°/80°	40%
SUN JUL 28	Scattered Thunderstorms	88°/79°	40%

- **Amplitudes can be positive or negative. They can be real or complex**

  This is valid… $|\psi\rangle = \frac{1}{2}|0\rangle - \frac{i\sqrt{3}}{2}|1\rangle$ **if measure we'll see** $P(0) = \frac{1}{4}; P(1) = \frac{3}{4}$

Fermilab

Interpretations of Quantum Mechanics (a fun or frustrating aside)

- **When no one is looking $|\psi\rangle \to U|\psi\rangle$; Measurement $|\psi\rangle \to i$ with probability $|\alpha_i|^2$**

- **Copenhagen Interpretation (Bohr & Hisenberg)**
  The quantum and classical worlds are different. The wave function collapses on measurement

- **Many Worlds (Everett)**

$$\frac{|😺\rangle + |🙀\rangle}{\sqrt{2}} |\,\textbf{You}\rangle \to \frac{|😺\rangle\,|\,\textbf{You}_0\rangle + |🙀\rangle\,|\,\textbf{You}_1\rangle}{\sqrt{2}}$$

  World branches upon observing (interacting with you). Multiverse!!

- **Dynamical Collapse, GRW, Penrose Theory**
  QM is not complete … maybe there's a rule that says that "big" systems collapse

- **Shut up and calculate! (David Mermin, Scott Aaronson)**
  The interpretations do not change the results of QM, so who cares! Only care about the answer to your problem

Quantum Interference

The fact that amplitudes can be positive or negative and real or complex means that states can interfere. Famous example is the double slit experiment

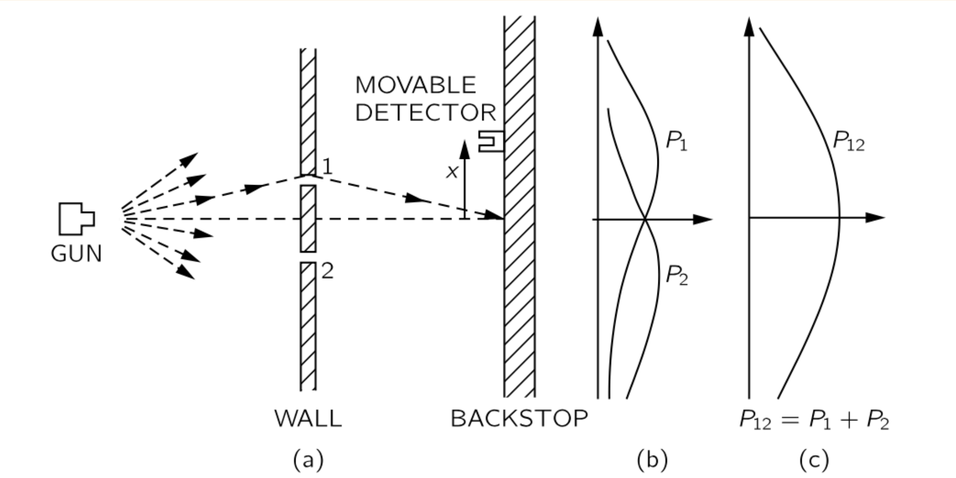


Fig. 1–1. Interference experiment with pellets.

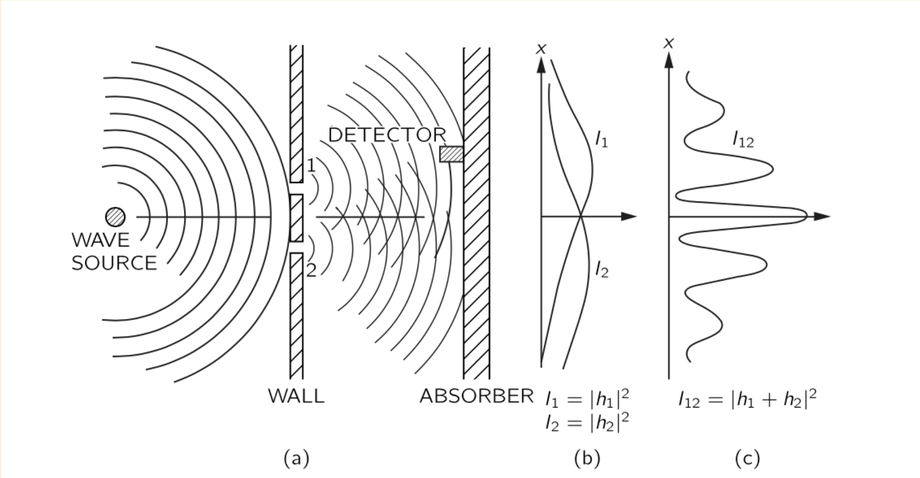


Fig. 1–2. Interference experiment with water waves.

http://www.feynmanlectures.caltech.edu/III_01.html#Ch1-S1

Fermilab

What do electrons do?

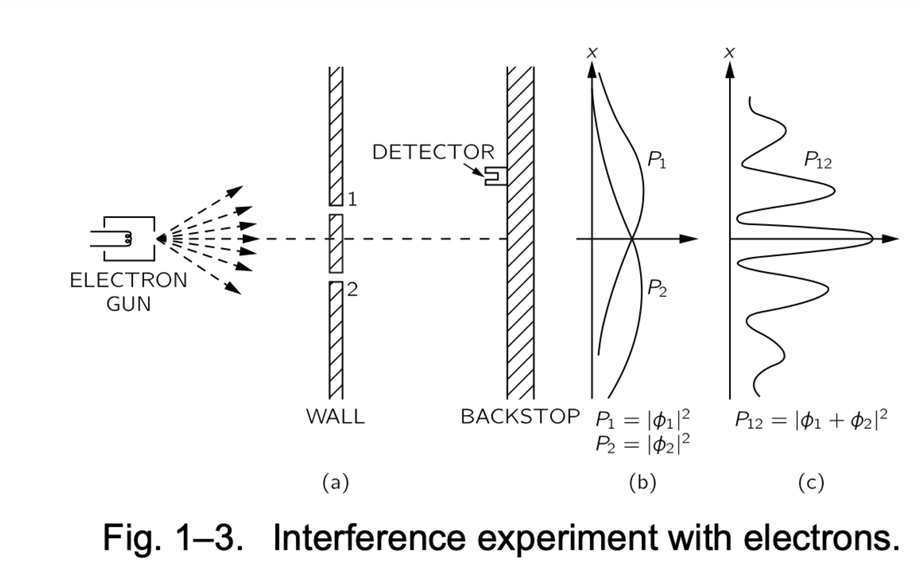
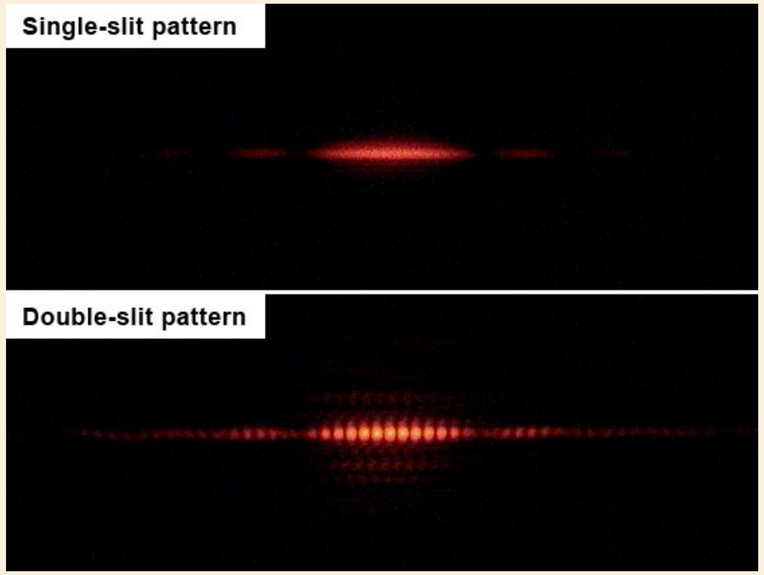


Fig. 1–3. Interference experiment with electrons.



https://en.wikipedia.org/wiki/Double-slit_experiment

- **Why do electrons behave like waves for double slit?**
  Two paths to a spot on the backdrop. Amplitude for upper and amplitude for lower
  $P(\text{hit}) = |\,\text{upper slit} + \text{lower slit}\,|^2$

- **Since amplitudes may be positive or negative, on some places on the backdrop they reinforce and on other places they cancel … make fringes … *the electron interferes with itself***

- **Note - if you watch the electron go through the slit, you get the "pellet" pattern, not the quantum pattern**

Fermilab

Huh?

- This all very strange … but we know this is how the world of electrons, protons, neutrons, atoms, molecules, proteins, etc works

- It is the “operating system” of the universe [Scott Aaronson]

- So - what does this have to do with computing?

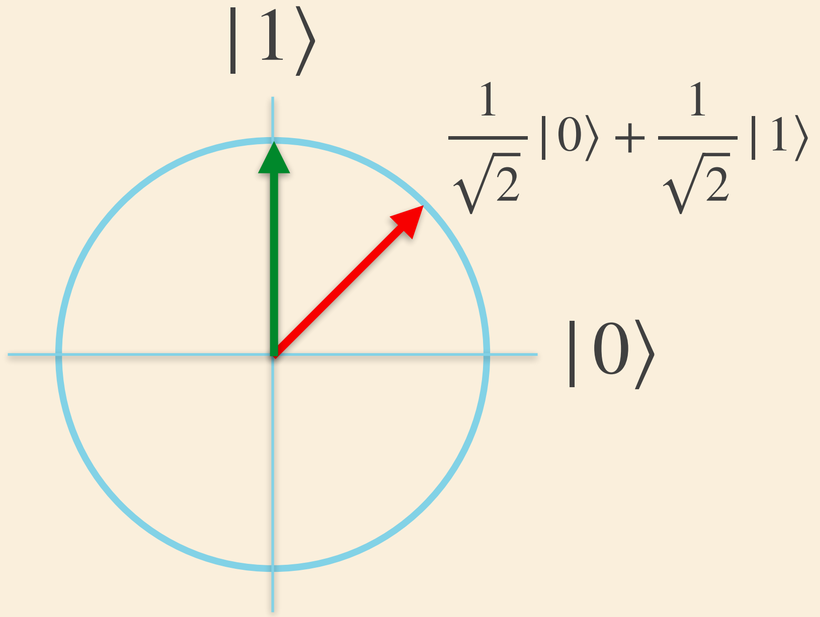
- Classical computers use “bits” - 0 or 1
  Perform operations on the bits (gates)… NOT, AND, OR, XOR, ADD, …

- Quantum computers use “qubits” … a two state system $|0\rangle$ and $|1\rangle$

Qubit

Unlike a bit, a qubit can be in a SUPERPOSITION of $|0\rangle$ and $|1\rangle$

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



When we observe (or measure) the qubit, we see it in either $|0\rangle$ or $|1\rangle$ with probability given by the amplitudes

e.g. For green, we will always observe $|1\rangle$

e.g. For red, we'll see $|0\rangle$ with 50% prob and $|1\rangle$ with 50% prob

For red, if we do this "experiment" many times, we'll see $|0\rangle$ half the time and $|1\rangle$ half the time

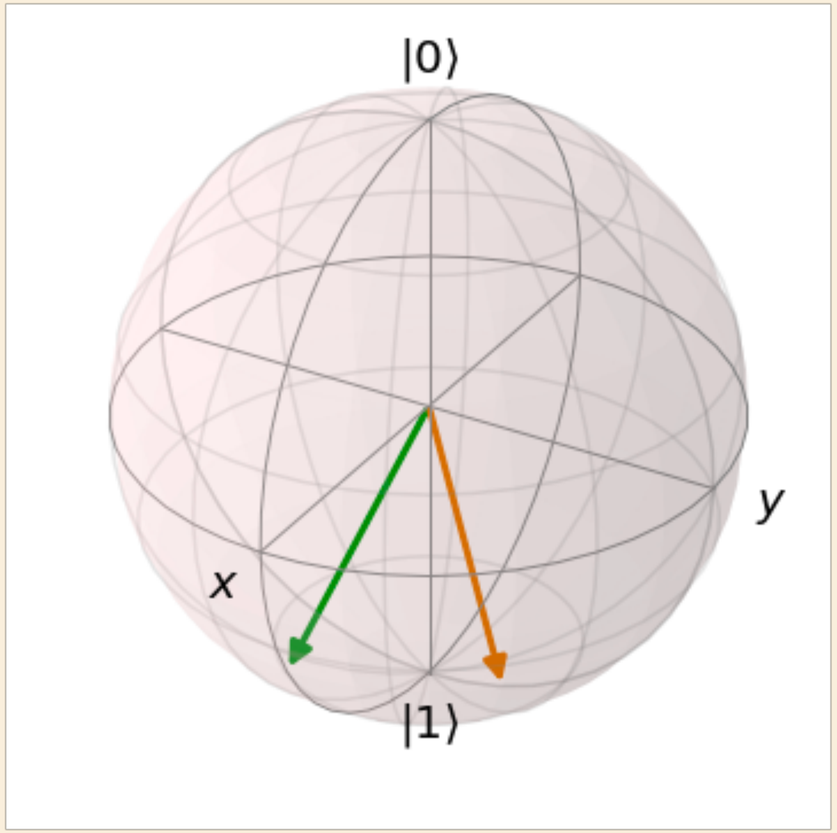
Fermilab

Bloch spheres

Actually, the circle representation is simplistic. Since amplitudes can be complex, there's another dimension… we use a "Bloch" sphere

$$|\psi\rangle = \frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle; \quad P(0) = \frac{1}{4}; P(1) = \frac{3}{4}$$

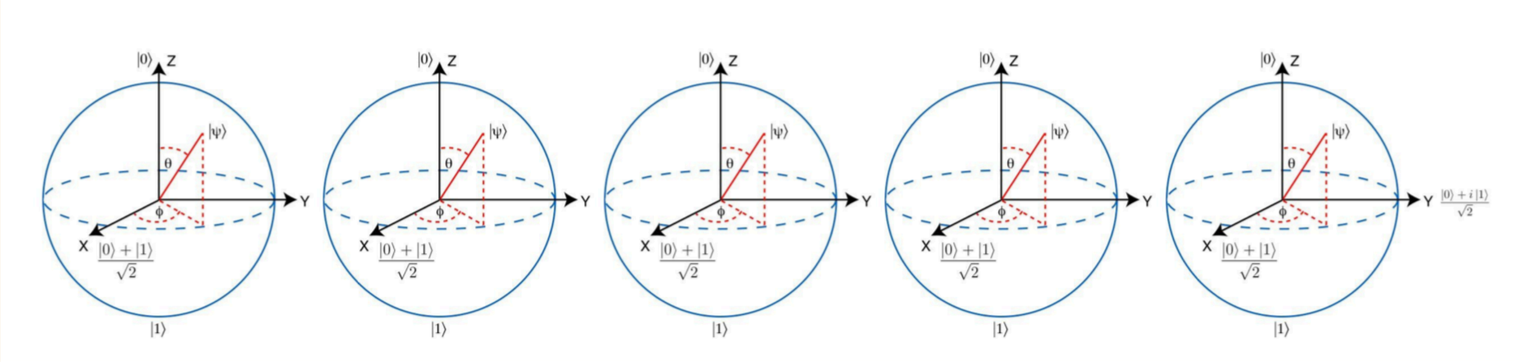
$$|\psi\rangle = \frac{1}{2}|0\rangle + \sqrt{\frac{3}{8}}(1+i)|1\rangle; \quad P(0) = \frac{1}{4}; P(1) = \frac{3}{4}$$



Fermilab

Quantum Information

You can have many qubits (IBM has a 49 qubit machine; Google has a 72 qubit)



...

Wave function is the sum of all states each with an amplitude

$$|\psi\rangle = a_0|0\ldots00\rangle + a_1|0\ldots01\rangle + a_2|0\ldots10\rangle + \ldots + a_k|1\ldots11\rangle$$

e.g. Three qubits $|\psi\rangle = \alpha_0|000\rangle + \alpha_1|001\rangle + \alpha_2|010\rangle + \alpha_3|011\rangle + \alpha_4|100\rangle + \alpha_5|101\rangle + \alpha_6|110\rangle + \alpha_7|111\rangle$

For n qubits, there are 2^n amplitudes (so $k = 2^n$)

To write down the wave function for a 49 qubit computer, you need 2^{49} = 562,949,953,421,312 complex numbers.

Fermilab

Quantum Information

My laptop has 32 GB or 2^{35} bits of memory

The world's biggest supercomputer (Summit at Oak Ridge National Lab in TN) has
> 10 PB or 2^{53} bits of memory

A 49 qubit Quantum Computer can "hold"
about as much information as the world's largest supercomputer

A 72 qubit Quantum Computer can "hold" about a million times more information than Summit

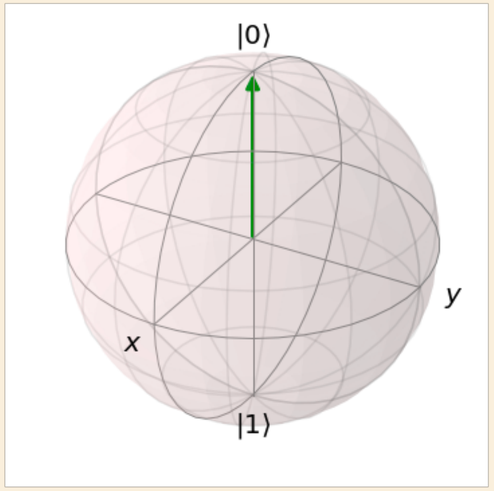
VERY difficult to simulate a Quantum Computer on a Classical Computer
(even a supercomputer - though with lots of tricks ~100 qubits have been simulated)

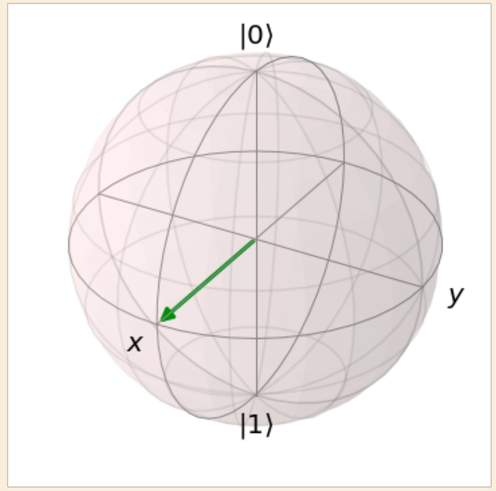
Fermilab

Quantum Computer Operations

Qubits can evolve and be manipulated without measuring them (gate operations):

- **Rotate state in the Bloch sphere … for example put qubit in a *superposition***

$|0\rangle$ 

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

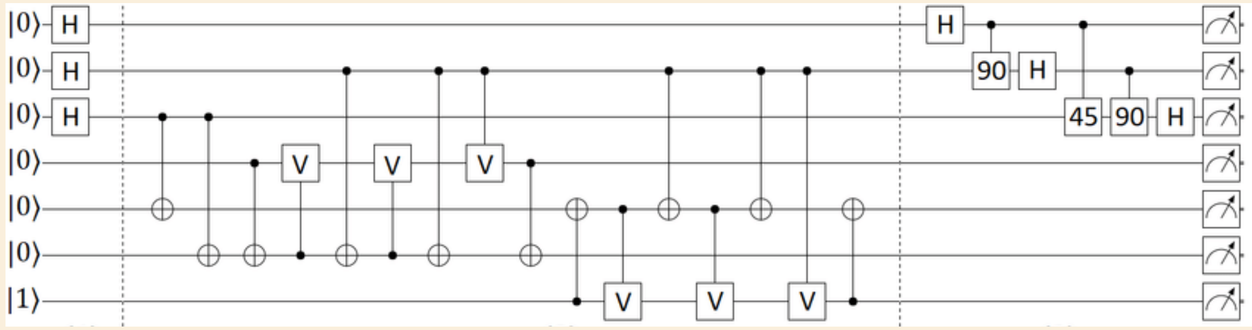
- **Entangle 2 or 3 qubits (e.g. make them dependent on each other)**

- **Measure one, some, or all of the qubits in the computer**

Fermilab

How quantum algorithms work — General principals

- Put the qubits in a superposition representing all the answers (right answers and wrong answers)



- If you measured the qubits now, you would randomly get a right answer or a wrong answer … that doesn't help! [Parallel computing in multiple universes is a misconception]

- Exploit Quantum **Interference**: Manipulate the qubits to "*choreograph the amplitudes*" [Scott Aaronson] so that paths to the wrongs answers cancel out and paths to the right answers reinforce

- Hopefully, when you measure the qubits, the probability of observing the right answer is high and probability of observing a wrong answer is low

- In practice this is very difficult and algorithms are very complicated — almost always named after the inventor — if you want to try an algorithm, go to https://github.com/lyon-fnal/qc-tutorial-fnal

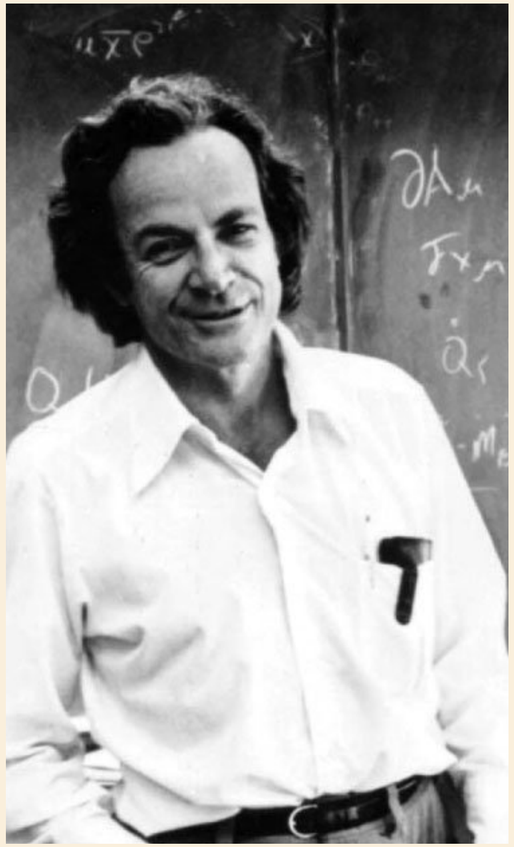
Fermilab

So what's all this good for?

- This is all amazing … and is how Nature WORKS!
- But aside from factoring numbers, what can you do with it?

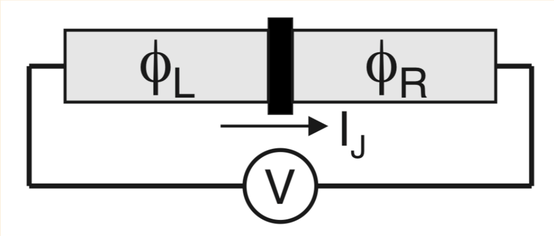
- Richard Feynman (1981) was one of the originators of QC:
  "…because nature isn't classical, …, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy"

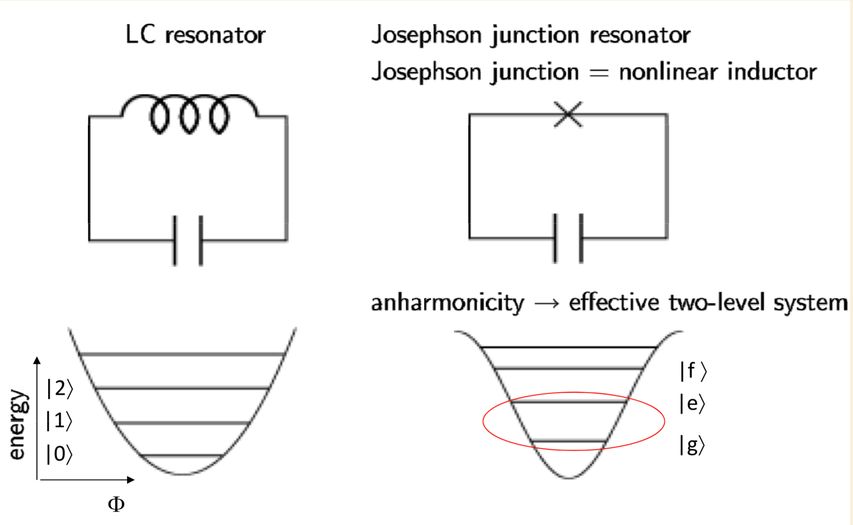
- Qubits are like artificial atoms that can be controlled. Simulate nature in a quantum mechanical way

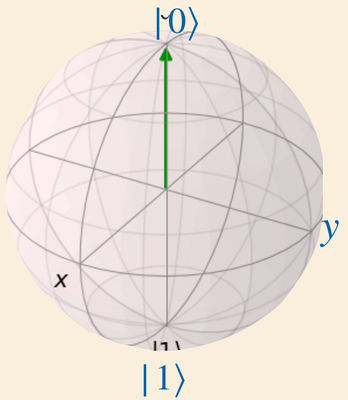


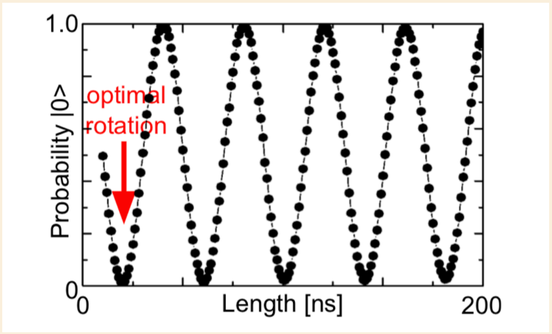
How do we make a quantum computer?

- Requirements
  - Qubits need some kind of physical representation and maintain quantum properties
  - We must be able to manipulate their quantum evolution (e.g. a transistor works by QM, but isn't a qubit)
  - We must be able to prepare their initial states and measure their final states
- Superconducting Qubits (artificial atoms)
  - Super-current tunnels through barrier between two superconductors
  - Combined with a capacitor — make a resonator
  - Josephson junction provides non-linearity to make anharmonic oscillator — usable 2 level system!
  - Microwave pulses rotate qubit about the Bloch Sphere:
    - Frequency is energy level difference, Axis selected by amplitude modulation, Angle set by pulse duration



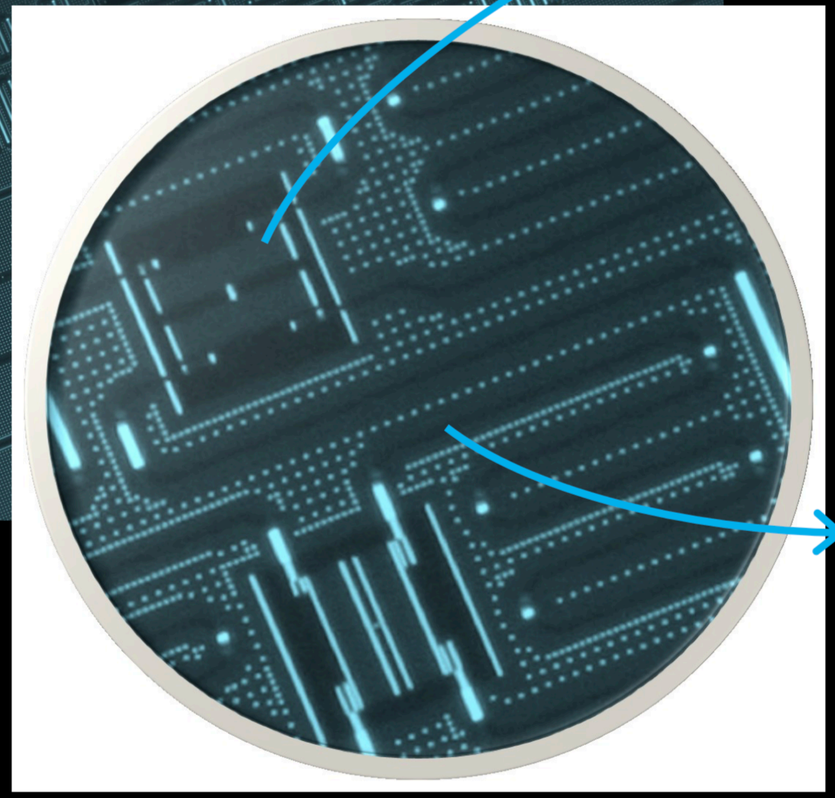






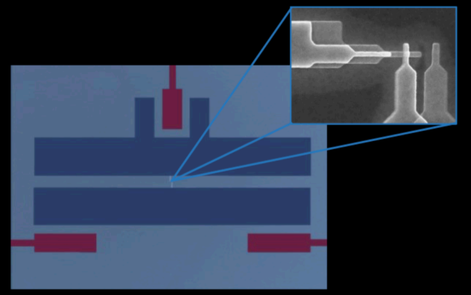
Fermilab

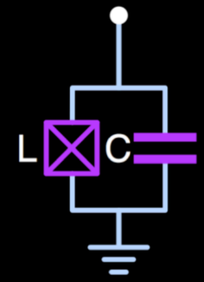
What do they look like (deep on the inside)?



Superconducting qubit

- quantum information carrier

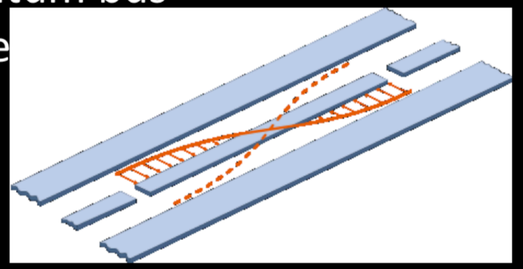


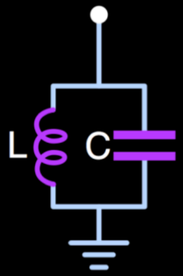


$E_{01} \approx 5\ \text{GHz} \approx 240\ \text{mK}$

Microwave resonator:

- read-out of qubit states
- quantum bus
- noise





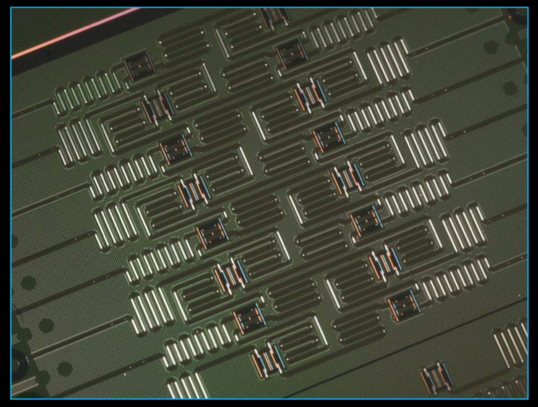
Fermilab

What do they look like (deep on the inside)?

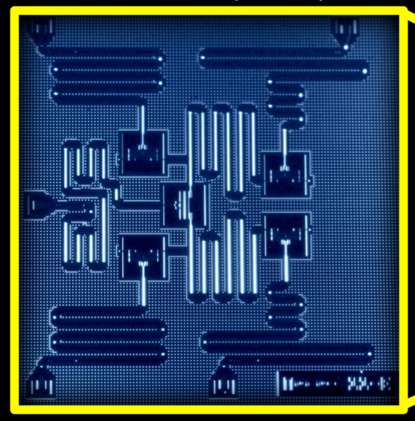
IBM qubit processor architectures

IBM Q experience (publicly accessible)

16 Qubits (2017)

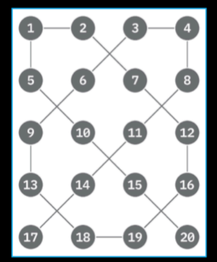


5 Qubits (2016)

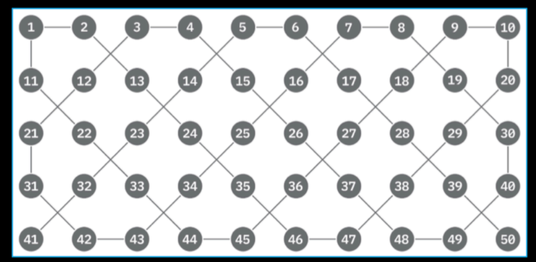


IBM Q commercial

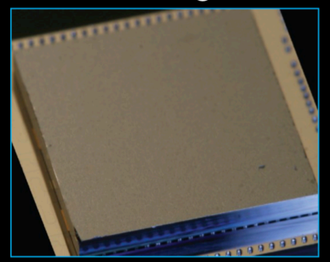
20 Qubits

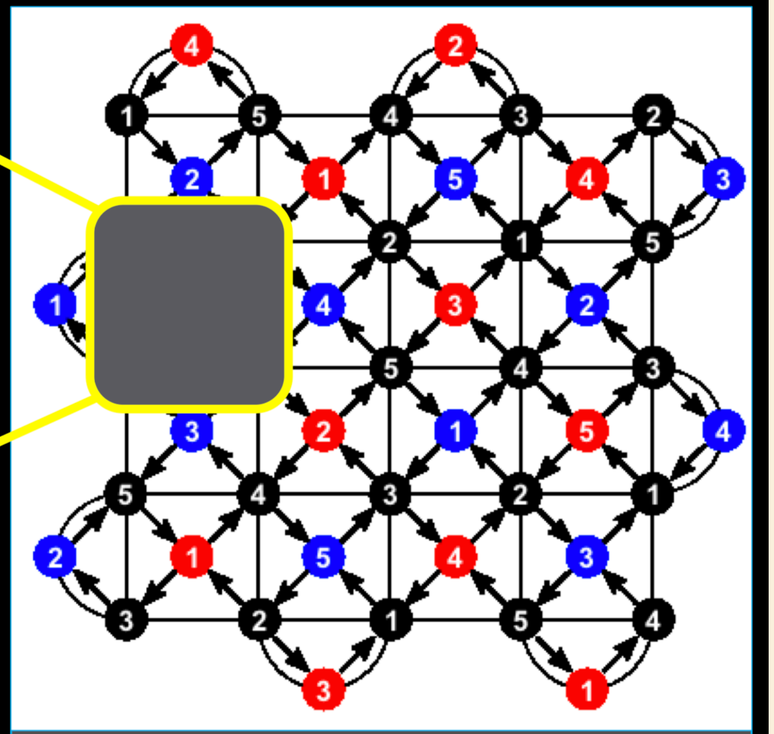


50 Qubit architecture (2017)



Package

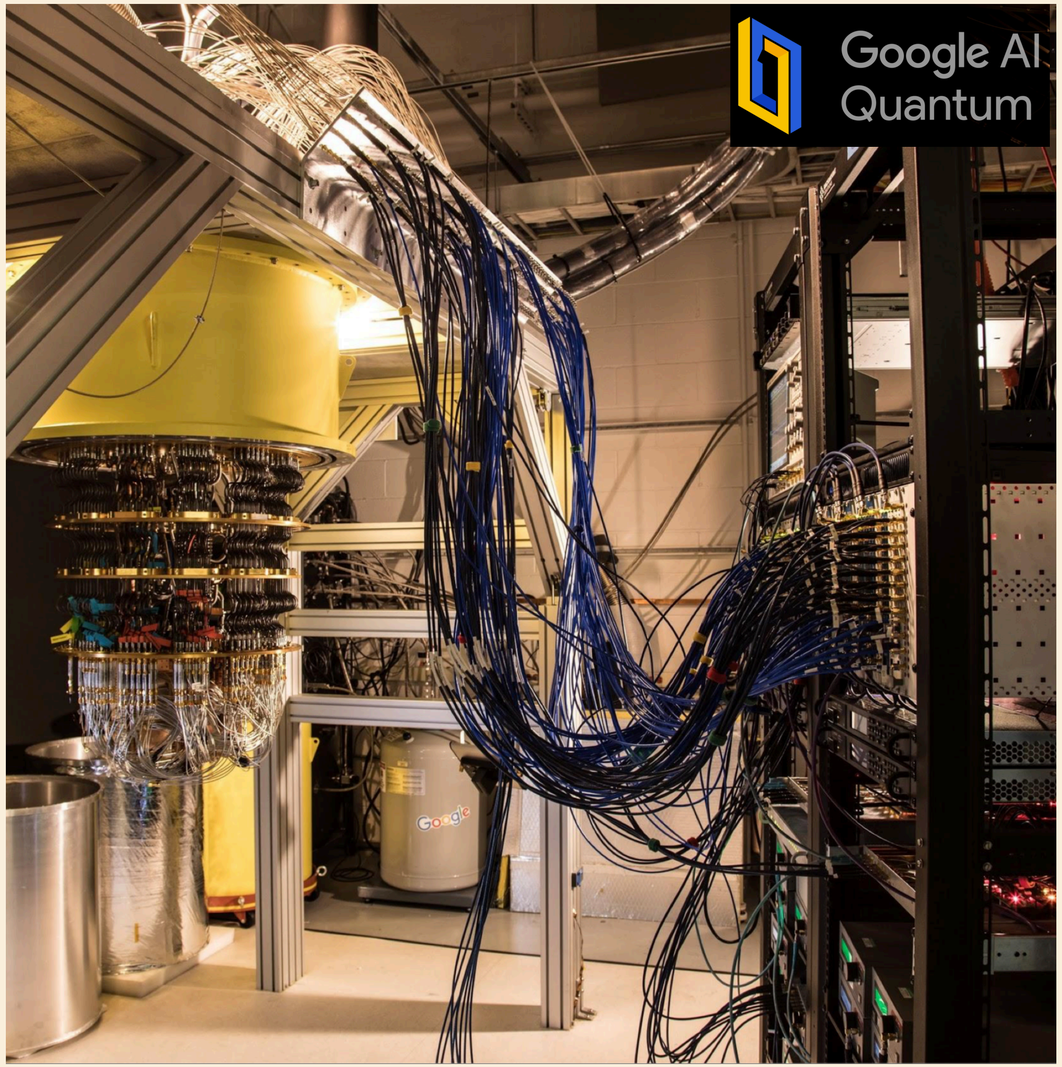




Latticed arrangement for scaling

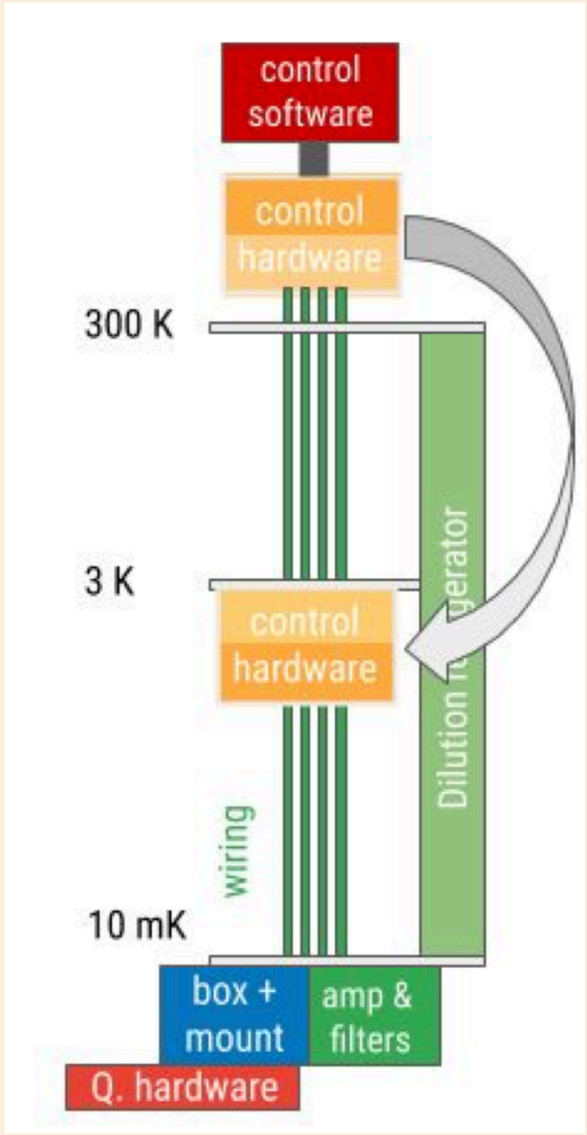
Fermilab

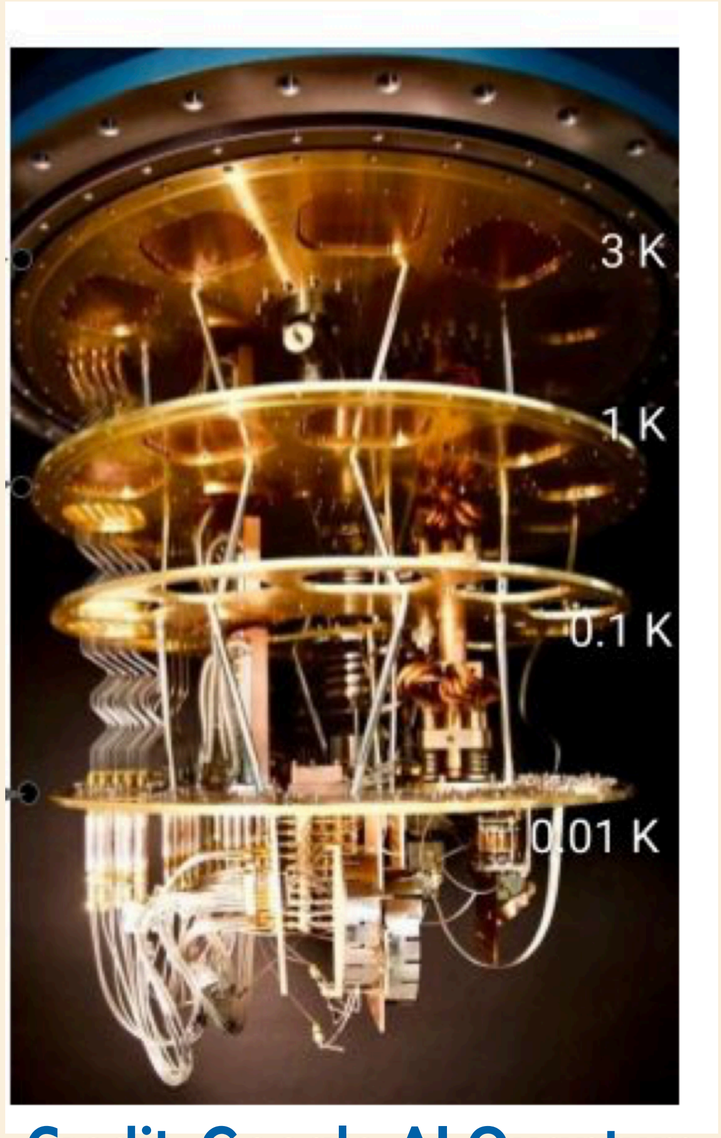
What do they look like (on the outside)?



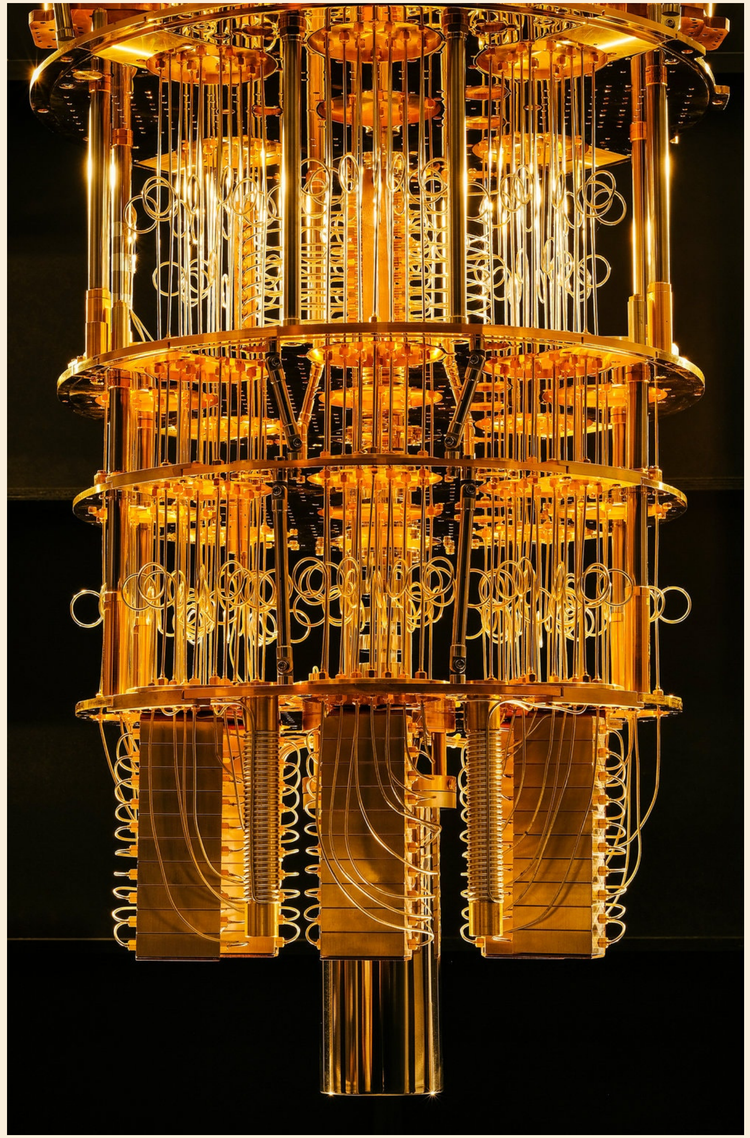
Fermilab

What do they look like (under the hood)?



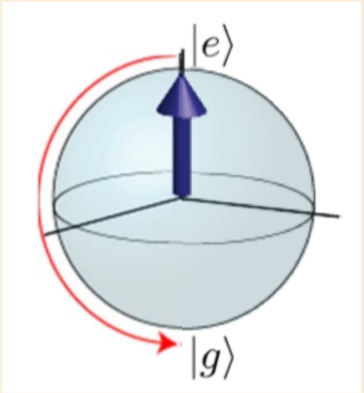


Credit: Google AI Quantum



Fermilab

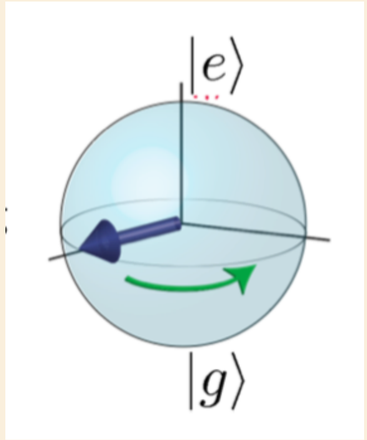
Limitations of Quantum Computers



All qubits have limitations:

Qubit “relaxes” to the lower energy state

~ 60 micro-seconds



Any contact with the outside world (e.g. heat) “dephases"

~ 100 micro-seconds

Fidelities (errors)

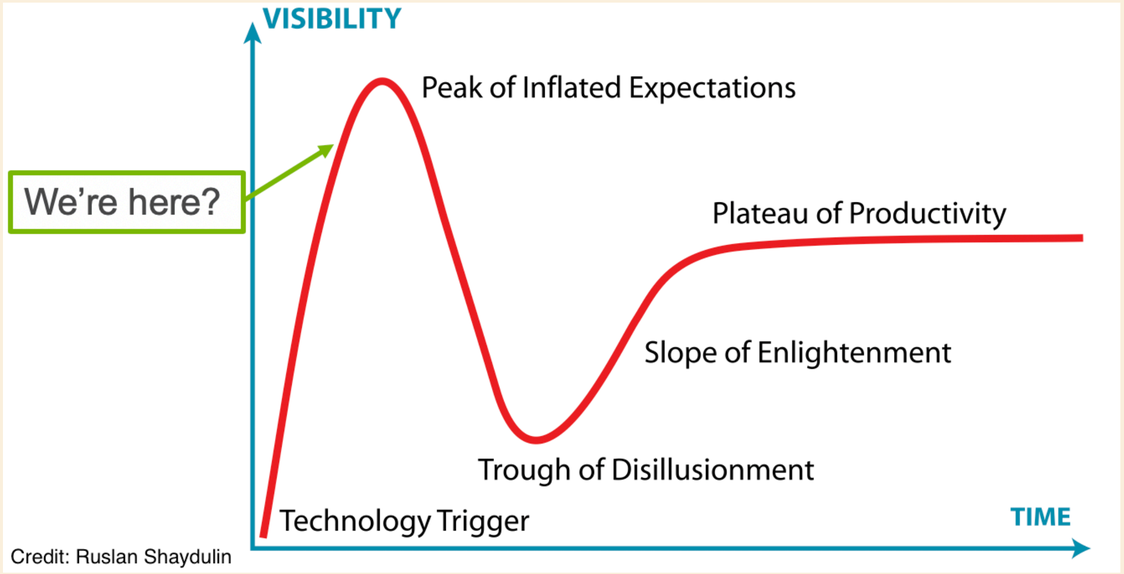
The qubits don’t always do what you tell them to (e.g. 99.8% gate fidelity)

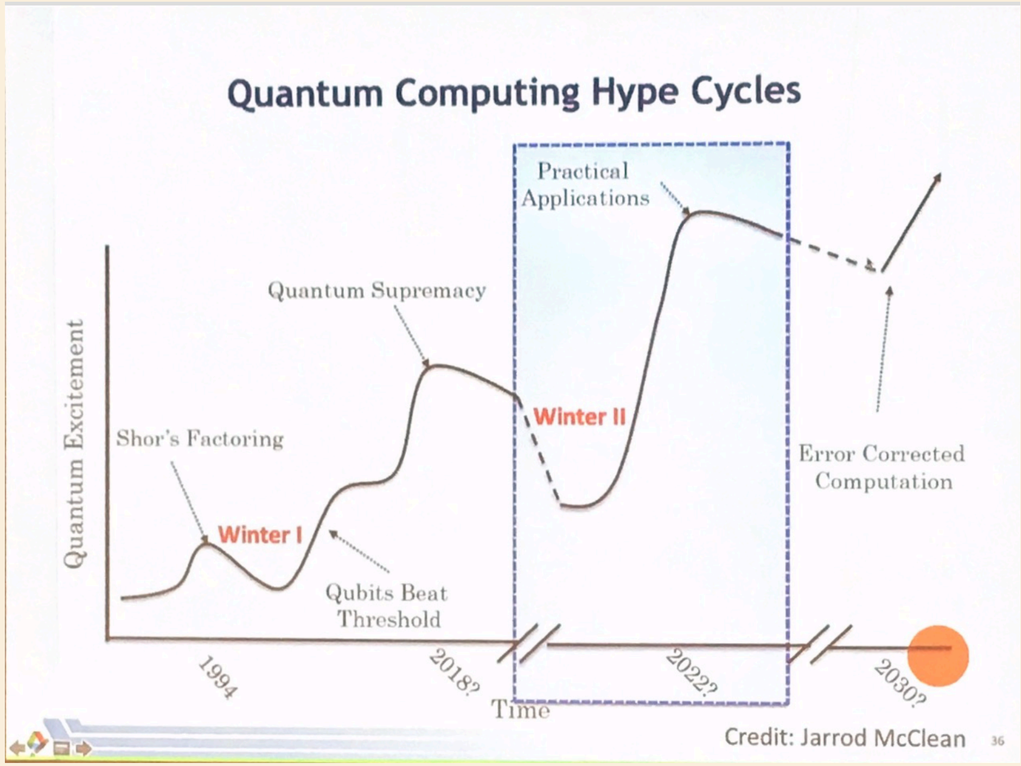
These limitations impact how much an algorithm can do!

Looking for error corrections, fault tolerance, and ways to increase times above

Fermilab

Where are we now?





- We're in the Noisy Intermediate-Scale Quantum (NISQ) era
  Current state: IBM has 49 qubits, Google has 72 qubits
- Other technologies are in development (ion traps, silicon spin, topological)
- Usefulness of quantum computers is to learn more about quantum computing
- Future will require millions of qubits … scaling and fault tolerance!

Fermilab

QC Applications

- Quantum Supremacy - may reach soon?
- Destroying encryption (it turns out there are Quantum-proof means of encryption)
- Optimization problems (scheduling, routing, operations) [Car & Aerospace Industry]
  - Finance, Quantum Chemistry, Forecasting, Machine Learning [Pharmaceutical, Cosmetics]
  - Learning more about Quantum Mechanics, Quantum Gravity, Blackholes, Wormholes

These applications are a long way away!

We just now have “baby” NISQ Computers that kinda do stuff. They need to scale up dramatically to be useful. Real applications will take ~ 1M qubits

Summary

Quantum Computers are a completely different type of computing machine

You won't use them to balance your checkbook or surf the web

(Just like you wouldn't light your dining room with lasers)

QC exploits properties of Quantum Mechanics

Seems amazing and magical, but this is how nature works and we know it!

QC is just getting started

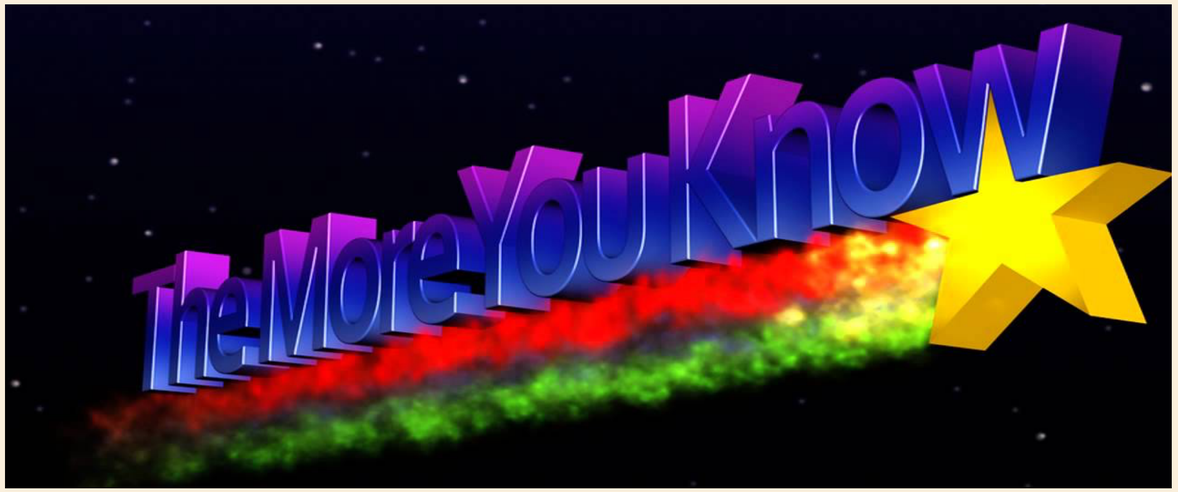
While the field has been around for decades, technology is catching up and we now have real quantum computing machines. But it is still early days and useful QC is likely at least a decade or more away.

Research has reached the point where it needs the management, organization, and resources of the national laboratories, so DOE, Fermilab, Argonne, and other national labs are getting involved

Why Fermilab? Particle Physics is built on Quantum!

Fermilab is contributing to Quantum Science with Superconducting Cavity Qubits, Microwave control electronics, particle detectors with qubit cavities, and classical computing expertise like analysis and simulations

Learning more



We've barely scratched the surface

Places to go to learn more:

- **Wikipedia articles on QC are quite good**

- **Scott Aaronson (UTA professor)** https://www.scottaaronson.com **and his talks on YouTube**

- **IBM Quantum Experience (actually play with a Quantum Computer in the cloud)** https://www.research.ibm.com/ibm-q/

- **If you can code in Python then see SDKs from Google, IBM, and Microsoft … they have lots of examples to play with (use simulations or real Quantum Computers in the cloud)**

- **Shor's algorithm is *very complicated*. Learn about it at "Minute Physics" videos at** https://www.youtube.com/watch?v=lvTqbM5Dq4Q **and** https://www.youtube.com/watch?v=FRZQ-efABeQ

Fermilab

And you?

Quantum Information Science is an exploding multi-disciplinary field

- Physics (Theory, Particle, Nuclear, Condensed Matter)
- Computer Science (including theoretical)
- Engineering (Electrical, RF, Microwave, Mechanical)
- Materials science

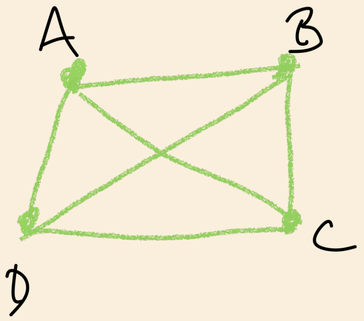
What places do Quantum Computing?

- Universities (many are starting Quantum Information Science/Engineering groups/departments)
- National Laboratories (all the labs are getting into this) … US is funding $1.2B with NQI
- Industry
  - Building Quantum Computers … IBM, Google, Microsoft, Intel, Regetti
  - Using Quantum Computers … Ford, Mercedes, VW, Boeing, Airbus, Amazon, …

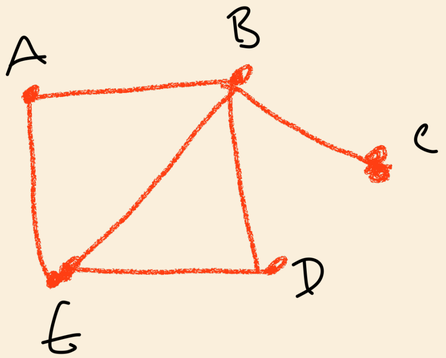
A fun time to get in … rapid advances are perhaps coming soon … a LOT of money is being spent

Backup Backup Backup Backup Backup Backup Backup Backup

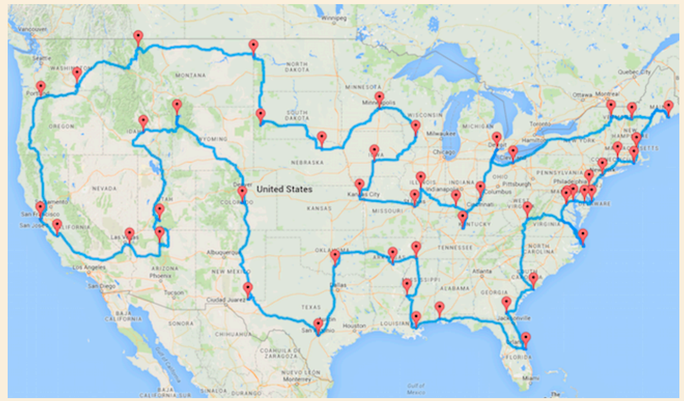
An NP problem - Traveling Salesman



- Can you make a closed loop, following connections (lines), visiting each node (dot) exactly once? (yes or no)



- E.g. 15 fully connected nodes has 43,589,145,600 possible routes. # of routes for 50 nodes is a 63 digit number
- Imagine millions of fully connected nodes
- Hard to solve, but efficient to verify
  NP & NP-hard & NP-complete



http://www.math.uwaterloo.ca/tsp/usa50/index.html

- Note - Finding the *shortest* route is *not* NP !
  - It is hard to solve <u>and</u> *not-efficient* to verify
  - NP-hard
    https://eklitzke.org/the-traveling-salesman-problem-is-not-np-complete

- Optimization problem - industrial processes, scheduling, etc

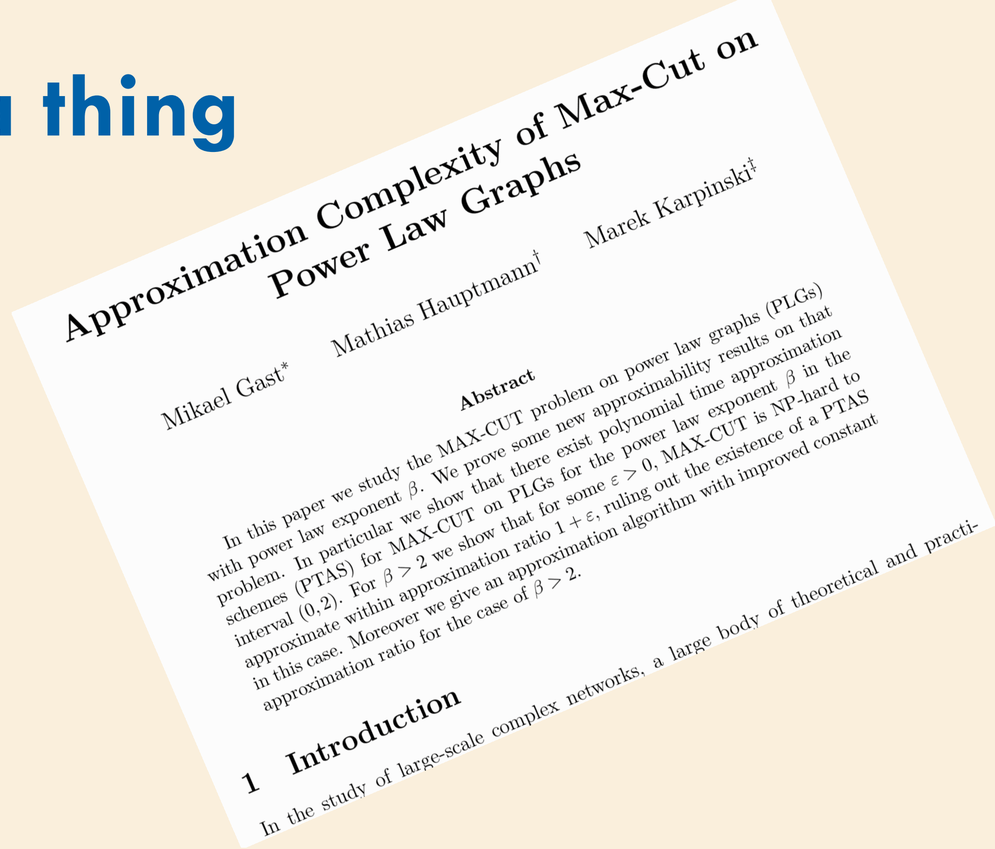
Fermilab

Note: Theoretical Computer Science is a thing

- You can make a career thinking deeply about algorithms mathematically (and not actually running them)

- Huge question: Does **P = NP ?**
  That is, for every NP problem, does a P solution exist?

- Computer Science Theorists are not willing to eliminate that possibility, despite reality



Approximation Complexity of Max-Cut on Power Law Graphs

Mikael Gast* Mathias Hauptmann† Marek Karpinski‡

Abstract

In this paper we study the MAX-CUT problem on power law graphs (PLGs) with power law exponent β. We prove some new approximability results on that problem. In particular we show that there exist polynomial time approximation schemes (PTAS) for MAX-CUT on PLGs for the power law exponent β in the interval $(0,2)$. For $\beta > 2$ we show that for some $\varepsilon > 0$, MAX-CUT is NP-hard to approximate within approximation ratio $1+\varepsilon$, ruling out the existence of a PTAS in this case. Moreover we give an approximation algorithm with improved constant approximation ratio for the case of $\beta > 2$.

1 Introduction

In the study of large-scale complex networks, a large body of theoretical and practi-

Fermilab