Jumpstarting Quantum Computing in the Middle and High-School Classroom: A Guide for Teachers and Learners

Experiential Learning

Catching a frisbee is not easy but both dogs and Computer Science (CSCI) sophomores seem to be good at it. How they actually do it is still very much subject for debate [6, 7]. That they might be calculating trajectories in real time, using Newton's equations, remains a very unlikely hypothesis. And yet it is undeniable that catching a frisbee demonstrates a working knowledge of Physics. How is this knowledge acquired? It seems safe to say that neither dogs nor CSCI sophomores learn their frisbee physics in the classroom. Both groups are comprised of very intelligent individuals but most individuals in these groups have a clear preference for experiential learning techniques (i.e., learn by doing) not to mention that their relatively short attention span sometimes presents a real challenge. Recently the concept of "embodied heuristics" [6] has been proposed as a possible operational explanation. Such a heuristic is a distillation of evolved sensory and motor abilities and is the result of practice and evolution. We conclude that if you want to be good at catching a frisbee you need to practice. And if you keep at it, you get good at it. Catching a frisbee is classical physics. Classical physics is all around us. Interaction with is is inevitable, ubiquitous, vital and fun.

Building an Intuition

Our experiences and the basic nature of systems that obey classical mechanics allow us to develop a working intuition for the behavior of many things we see about us (and interact with) daily. But, classical physics is a limiting case of quantum physics and as Dirac taught us, there is a minimum disturbance that accompanies a measurement, a disturbance that is inherent in the nature of things and can never be improved by experimental technique. "If the disturbance is negligible, then the object is large in an absolute sense, and it can be described by classical physics. However, if the minimum disturbance accompanying a measurement is nonnegligible, then the object is absolutely small, and its properties fall in the realm of quantum mechanics. The quantum properties of absolutely small particles are not strange; they are just unfamiliar and not subject to our classical intuition." [4] Thus it may be accurate to say that a quantum object is produced as a particle, propagates as a wave and is detected as a particle with the probability distribution of a wave but what difference does that make to a computer scientist? Why do we need quantum mechanics?

Quantum Computing

The UN has named 2025 the International Year of Quantum Science and Technology on the occasion of 100 years of Quantum Mechanics. Quantum computers harness quantum mechanics to compute by different rules than classical computers do. They don't perform operations faster than a classical computer but they perform different operations that a classical computer can't, and sometimes those operations offer a faster route to a solution.

At the foundation of our field we have two rewriting systems: Turing machines and lambda calculus. Turing machines are important for many reasons, but especially because of two long-held beliefs regarding computation: first, the Church-Turing thesis says that everything that is computable can be computed with a Turing machine, although it could in some cases take a very long time (i.e., exponential time in the size of the input). This correctly suggests that there are problems that cannot be computed—they are called undecidable problems, the most famous of which is the halting problem. Aside from such uncomputable problems, everything else can be computed, and it can be computed using a Turing machine.

Second, the extended Church-Turing thesis is a(nother) foundational principle of computer science that says that the performance of all computers is only polynomially faster than a probabilistic Turing machine. In other words, any model of computation, be it the circuit model or something else, can be simulated by a probabilistic Turing machine with at most polynomial overhead. A probabilistic Turing machine is a Turing machine where the state of the system can be set probabilistically, such as by the flip of a coin. The strong (or extended) Church-Turing thesis says that a probabilistic Turing machine can perform the same computations as any other kind of computer, and it only needs at most polynomially more steps than the other computer. In 1993, Bernstein and Vazirani showed that quantum computers could violate the extended Church-Turing thesis. Their quantum algorithm offered an exponential speedup over any classical algorithm for a certain computational task called recursive Fourier sampling. Another example of a quantum algorithm demonstrating exponential speedup for a different computational problem was provided in 1994 by Dan Simon. Quantum computation is the only model of computation to date to violate the extended Church-Turing thesis, and therefore only quantum computers are capable of exponential speedups over classical computers.

It's equally important here to understand that quantum computers would not violate the regular Church-Turing thesis. That is, what is impossible to compute will remain impossible. The hope, however, is that quantum computers will efficiently solve problems that are inefficient on classical computers. One such problem is the factoring of very large numbers. Another one is simulating nature with computers. Nature appears to be following the laws of quantum mechanics. Quantum mechanics is complex and sometimes classical computers can struggle to crunch the numbers to figure out what nature is doing. But quantum computers play by different rules. Quantum computers don’t need to crunch these numbers per se, they can simply mimic nature rather than approximate it numerically like the classical computers need to. And that’s because, just like nature, quantum computers are quantum. And the potential here is enormous not just for understanding physics but for designing new materials, and medicines, for instance.

Student Agency

Having decided that the topic is important we now ask ourselves what learner-sighted teaching technique is best suited here (and in general). Student agency is the ability to manage and advance one's learning. What we want to see is the learner in pursuit of knowledge, not knowledge in pursuit of the learner (at all levels). Education should foster independent exploration and construction of knowledge, rather than passive acceptance of instruction. Though we agree that a motivated student will always be in pursuit of knowledge, all too often in school we find that knowledge is in fact in heavy pursuit of the student. Furthermore, we believe all students are intrinsically motivated to learn but learn to be unmotivated if they repeatedly fail. Every student has the basic needs to belong, to be competent and to influence what happens to them; motivation to learn only exists when these three conditions are satisfied. This is true at both the elementary level as it is in higher education. With this in mind we have developed and present an operational approach to jumpstarting quantum computing education to learners as early as middle school (or HS). Here we will restrict ourselves to present the phase kickback phenomenon and the Bernstein-Vazirani algorithm using just the basic rules of arithmetic. Our approach is based on a string-rewriting system invented by Terry Rudolph and introduced in his 2017 book "Q is for Quantum" [10, 11, 12]. We start from classical bits and little by little we introduce phase, superposition, interference. We show the simple rules that can help a middle school student trace qubits through a quantum circuit. We show how to verify what we do using the misty states formalism with circuits implemented in Qiskit. The reader is invited to read along with a pen and some paper. A laptop would come in handy as well.

Misty States

The 12-year-olds of today may well have access to large quantum computers before they leave their teenage years. Yet a standard educational trajectory would see them still several years away from learning enough quantum theory to explore this technology's amazing potential meaningfully. In addition to barriers of convention ("This is the order in which things have always been taught") there are math-related barriers ("You can't understand quantum theory until you have mastered linear algebra in a complex vector space"). But, as has been shown, and in true CSCI spirit, it is possible to replace linear algebra with some string-rewriting rules [10] which are no more complicated than the basic rules of arithmetic. These rules are very simple indeed but we have to warn the reader of underestimating them. In class we emphasize that mastery of any system, no matter how simple it may appear to be, requires both attention and practice. With these two conditions satisfied we're convinced that the reader will be very succesful.

It is important to note that our focus is quantum computing (QC) and not quantum mechanics (QM) or quantum physics in general. Learning QC is much easier [3] than learning QM because QC deals with a very simple subset of QM as follows: (a) a qubit—the foundation of quantum computing—is the simplest non-trivial quantum system; (b) you never have to solve the Schrödinger equation, or even learn what it is, because the quantum systems that carry out quantum computations evolve in a controlled manner based on the quantum gates applied to them; and (c) there's already a model of quantum computation, so the most difficult aspect of quantum mechanics—the art of applying it to real systems—is absent. We approach presentation from the mindset of maker-centered learning: "What I cannot create I cannot understand" is a good description of that persuasion and a quote from Richard Feynman. From here on, whether we discuss single or multiple qubit systems; entanglement; teleportation; quantum states, quantum gates and measurement; evolving quantum states with quantum gates; quantum circuits; primitives for a quantum processing unit; reversible computation and/or quantum algorithms, we advocate an environment of concrete representations via Python, Qiskit and the misty states formalism (the method developed and introduced by Terry Rudolph).

Maker-Centered Learning

According to Piaget "children in the early years of primary school need concrete[1] objects, pictures, actions, and symbols to develop mathematical meanings." The same is true of students who lack a certain background or affinity for the pure structures of mathematics. This is where the simplicity of the misty state formalism shines through. Piaget also said "[l]ogic and mathematics are nothing but specialized linguistic structures." The misty state formalism can facilitate access to both. Another quote, from Seymour Papert, is relevant here: "If people believe firmly enough that they cannot do math, they will usually succeed in preventing themselves from doing whatever they recognize as math." The consequences of such self-sabotage is personal failure, and each failure reinforces the original belief. Papert also said: "My basic idea is that programming is the most powerful medium of developing the sophisticated and rigorous thinking needed for mathematics." So our approach is trying to scaffold the knowledge needed to understand quantum computing and quantum information science starting from computing in Python in a notebook (Google Colab). We build an understanding of the misty state formalism and then use it to define, recognize and synthesize (operationally, in Qiskit/Python) the following concepts: superposition, phase, interference, entanglement, quantum gates and quantum circuits, the Deutsch-Josza algorithm, the Grover search algorithm, the Bernstein-Vazirani algorithm (and the phase kickback phenomenon that makes it possible) along with superdense coding and the GHZ game (quantum pseudo-telepathy via quantum entanglement). We then need to extend the system and present quantum teleportation and the phenomenon known as entanglement swapping (which allows qubits that have never met to become entangled). In this paper we only present the Bernstein-Vazirani algorithm via phase kickback and misty states. The rest has been presented and is available elsewhere and is now essentially part of the CS2023 report as a separate knowledge unit (KU).

Quantum Flytrap

We emphasize again that our central goal is not quantum mechanics. However we also need to stress that "[s]tudents and professionals interested in quantum information sciences need to adopt a different way of thinking than the one used to construct today's (classical) algorithms. This certainly presents tremendous challenges, since, for many years, computer science students have been led to believe that they can get by with some knowledge of discrete mathematics and little understanding of physics at all. [In quantum computing w]e are going back to the age when a strong relationship between physics and computer science existed." [8] Having said all of this we also need to point out that we don't consider detailed knowledge of QM a necessity for a CSCI student unless they decide to choose a career in building quantum computing hardware. Here (as is done in the CS2023 KU) we only promote an appreciation of (and familiarity with) the main quantum concepts: qubit, state, phase, interference, entanglement, teleportation, measurement, sensing, coherence, quantum communication and the main differences between QIS and QM. An environment facilitating direct interaction with these concepts is the Quantum Flytrap [2] which self-describes as a no-code IDE for quantum computing and we hereby strongly encourage its extensive use in the classroom and in labs.

---

[1] Our brains need to interact with something in order to create a model of it. As Papert puts it: "You can't think about thinking without thinking about thinking about something."

We will be giving two brief examples a bit later explaining why we think the use of such an IDE in the QC classroom (at all levels) is extremely beneficial.

In conclusion we consider experiential learning a *sine qua non* feature of learning for the kind of learners and topics we have in mind. In such a process building an intuition via embodied heuristics is fundamental but, as we have said, direct interaction with the world of the very small is expensive and it has to be mitigated since we're so big. John Preskill once remarked along the same lines: "Perhaps kids who grow up playing quantum games will acquire a visceral understanding of quantum phenomena that our generation lacks." With this in mind we advocate the use of Quantum Flytrap as a tool to complement the system developed and introduced by Terry Rudolph (misty states formalism) which we proceed to introduce next.

Bernstein-Vazirani

In presenting this simplified version of the celebrated algorithm we want to make very clear from the outset what is so remarkable about it. The problem states that we have a circuit in which we have placed a number of (quantum) gates. We will carefully define their kind and behavior shortly, along with their associated connectivity. The circuit will be presented to us as a black box. It will have a number of inputs and an equal number of outputs. We will be asked to determine the internal connectivity of the black box by just interacting with it from the outside. Using only classical physics (gates and principles) we conclude that the task of determining what the black box looks on the inside is linear in the number of inputs. But if we are allowed to use quantum physics (both hardware and principles) the same task can be solved in just one step, regardless of how many inputs the circuit has. Here's an example:

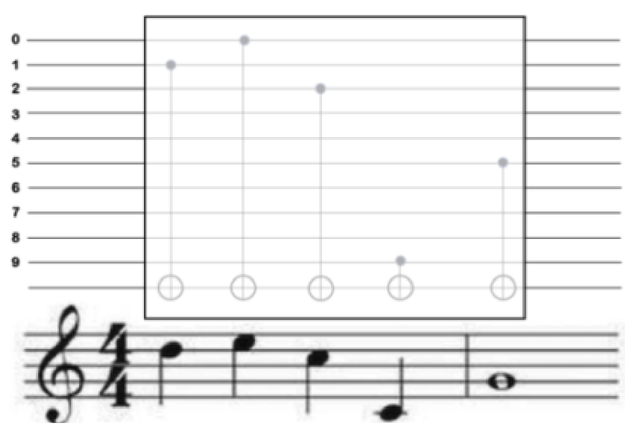


Figure 1. Bernstein-Vazirani challenge. Musical score is from "Close Encounters of the Third Kind" (Truffaut).

We now proceed to define the gates and the formalism we need. Familiarity with the material in the first part of "Q is for Quantum" is desirable but won't be assumed. As a result we will first introduce some of the material already in the book (NOT and C-NOT gates along with the Hadamard (PETE) gate) and the associated misty state formalism. We will then proceed to prove the phase kickback phenomenon and use it to solve the Bernstein-Vazirani challenge.

The NOT Gate

An excellent resource here is the 20-minute video [12] available on the book's [11] website.

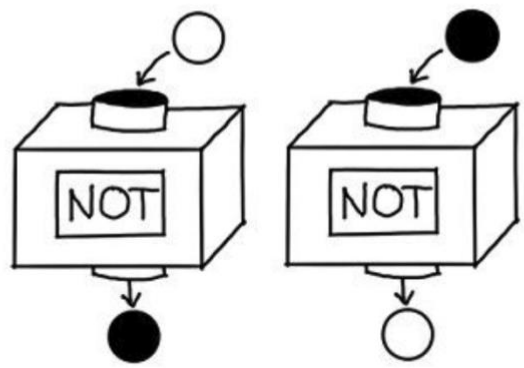


Figure 2. How the NOT gate works.

The classical bits are 0 and 1. We can represent them as W and B and draw them as a white or black ball. Indeed they are classical values. A quantum bit (qubit) is a more complex entity but when we measure a qubit we only get one of these two values, W or B. So we start from them. We define the behavior of the NOT gate as in the picture. The effect of the gate is consistent with our knowledge of (classical) logical gates: the NOT gate flips its input. We could also write NOT(W) = B and NOT(B) = W to describe what happens in these two pictures.

The C-NOT Gate

The controlled-NOT (C-NOT) gate has two inputs: a target and a control. It works by flipping the target when the control is a black ball. Here's the diagram from the book:

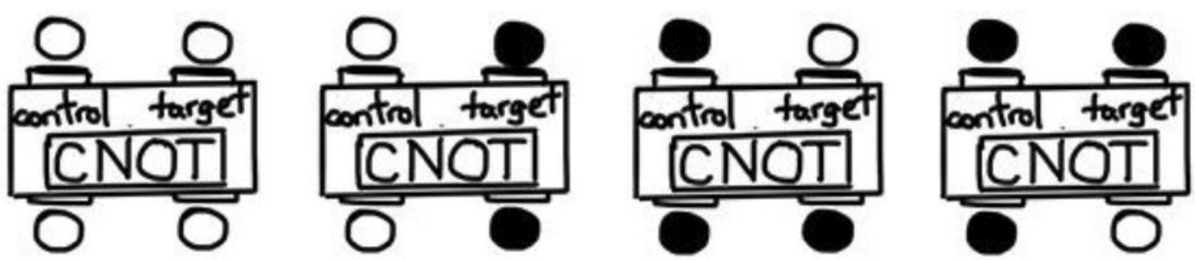


Figure 3. Behavior of the two qubit gate C-NOT.

A Simple Circuit

The next thing we consider is that by stacking boxes on top of each other, we can use the output of one box as the input to another. For example, we can stack two NOT boxes, with the result that the output now matches the input:

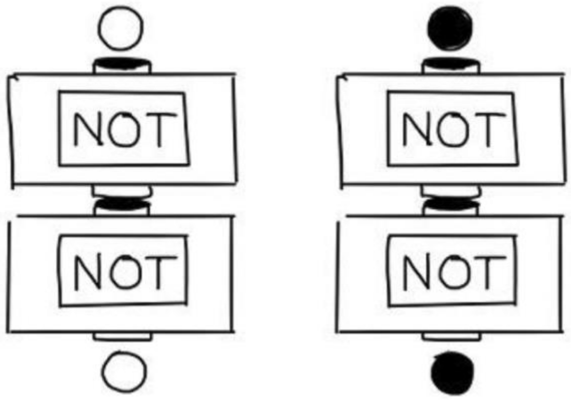


Figure 4. Stacking two NOT boxes recovers the initial input.

We can now set up a Google Colab session and write a simple Python program to solve a sample probability problem. The statement of the problem is: “You roll three dice, then sum the three outcomes, what is the probability of not getting a prime number?” Here’s the code and the answer:

```
[ ] count = 0

    for d1 in range(1, 7):
      for d2 in range(1, 7):
        for d3 in range(1, 7):
          sum = d1 + d2 + d3
          if sum not in [3, 5, 7, 11, 13, 17]:
            count += 1
          else:
            pass

    print(count/(6 * 6 * 6))

    0.6620370370370371
```

Figure 5. Calculating probabilities with Python in Google Colab.

Problem 1. Define fair as having an expected value of 0 (zero). Probability of rolling a 7 is $\frac{6}{36} = \frac{1}{6}$.The expected value $(4 \cdot \frac{1}{6} - 1 \cdot \frac{5}{6}) = \frac{4}{5} - \frac{5}{6} = -\frac{1}{6}$. Not fair. Fair would be with 5 instead of 4 for a win.

Figure 6. Another way of solving a probability question in Google Colab.

Another way of solving a probability problem in Google Colab is by writing the argument using LaTeX as shown in Figure 6. Here the problem is: “Craps is a game played with two dice. You decide to play. Each bet is one chip. The goal is to roll a seven (with two dice). The house pays 4 chips plus your original chip if you win. Is this fair? (If not, define fair).” A third way to solve such a problem is to have the computer run a simulation for you:

```
import random

times = 1000000
money = 0

for i in range(times):
  d1 = random.randint(1, 6)
  d2 = random.randint(1, 6)
  sum = d1 + d2
  # print(d1, d2, sum)
  if sum == 7:
    money += 4
  else:
    money -= 1

print (money/times)
```

```
-0.169725
```

```
import random

count = 0
times = 10000
for num in range(times):
  d1 = random.randint(1, 6)
  d2 = random.randint(1, 6)
  d3 = random.randint(1, 6)
  sum = d1 + d2 + d3
  if sum not in [3, 5, 7, 11, 13, 17]:
    count += 1
print(count/times)
```

```
0.6593
```

Figure 7. Simulating a game with two dice (left) and three dice (right).

```
[2] from qiskit import QuantumCircuit
```

```
[3] qc = QuantumCircuit(1)
    qc.x(0)
    qc.x(0)
    qc.draw(output='mpl')
```


```
from qiskit.quantum_info import Statevector
a = Statevector.from_instruction(qc)
a.draw('latex')
```

$|0\rangle$

Figure 8. Stacking two NOT boxes in Python (Qiskit). The input is recovered, as expected.

The assumption here is that the students already have some familiarity with Python and with the Google Colab interactive notebook environment. Access to the quantum emulator is immediate. Let's now introduce a most important quantum gate: the Hadamard gate.

A Necessary Detour

The Hadamard gate is a fundamental single-qubit quantum gate. We take a brief detour to discuss its role in quantum computing where it is used to create superposition states. In the book [10] it is known as the PETE box and the reason for why it's called that way can be deduced from the discussion below. Our goal in this paper is to introduce the misty state formalism from [10] in its pure form and use it to prove the phase kickback phenomenon. We have already mentioned all the gates we need (NOT, C-NOT and the Hadamard (PETE) box). We are in the process of defining (and using) the last of these three gates, the PETE box. At this point we want to stress why the misty state formalism, which will be used unchanged, that is, without any coefficients whatsoever, throughout this paper is so effective. It is true that one needs to extend this formalism to properly deal with phenomena such as W-entangled states and teleportation but that does not diminish the surprising effectiveness of the formalism as initially proposed in "Q is for Quantum." Without any changes to what is introduced in the book one can successfully present entanglement, Deutsch-Josza, Grover search, superdense coding and the GHZ game. To understand that part and how and when we need to provide the extension we quote from Terry Rudolph's FAQ on the book's website:

> "In the book I used only a single 'actually quantum' box, the PETE box. By this I mean it is the only box that has 'mist-creating' properties. All the remaining boxes introduced are things that just shuffle colors around—they would be at home in a classical computer for example. Only having to introduce a single new mysterious thing is very nice pedagogically. […] Now for the […] genesis of the whole misty-method: You may wonder whether my reliance on only the single PETE box is limiting, in the sense of […] does it limit the calculations you could do, and the phenomena you can demonstrate? […] The answer is that it is not limiting, that every calculation can be done (to good-enough accuracy, and again, perhaps with a small overhead) using only PETE boxes and the classical boxes. This is a remarkable mathematical result due to [Yaoyun] Shih, leveraging another powerful result (I think due to Kitaev). There is a citation at the end of the book. A few years ago I was in the middle of pondering this result when I realized I was running late to give a talk at a math camp for 12-14 year olds which was being run in part by my friend PETE Shadbolt. I raced for the tube, and while on it thought about what could I explain to these kids that wasn't the usual jargon-filled quantum fluff. And so here we are."

We said that later, after the content of the book [10] is mastered, we will need to (and we actually do) extend the original, "pure" misty state formalism. For us it happens (in a 6-8 week class entitled "Introduction to Quantum Advantage" [5] that also serves as a boot camp to our accelerated Master's program in QIS) when we try to implement W-entangled states which rely on the use of controlled-Hadamard gates and arbitrary qubit rotations. It happens again when we discuss teleportation, since the input to the quantum teleportation algorithm is an arbitrary[2] quantum state. Here's how this extension of the formalism is anticipated by Terry Rudolph in the FAQ of [10, 11] (where the previous quote came from):

> "[T]he misty formalism is 'universal', in as much as you can use it to do any quantum calculation with only a small overhead. I should reiterate I am not advocating that we should recast all of quantum theory into this formalism. The misty state picture is a good way of getting people to the heart of some nontrivial quantum theory without them having to absorb a boatload of irrelevant math. But that math is not largely irrelevant if you actually want to work in the field, it makes many things much easier."

Math, which is essential if you actually want to work in the field, because it makes many things much easier is our ultimate goal here as well. For example we'd like our readers to be able and ready to read [1, 9, 13] as soon as they master the contents of our class.

The PETE Box

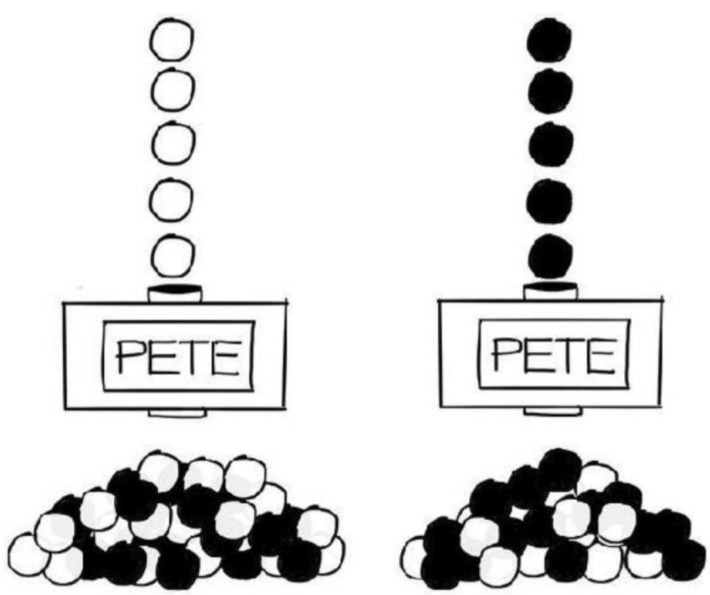


Figure 9. Behavior of the Hadamard gate (also known as the PETE box in [10]).

We capture the behavior shown in Figure 9 by introducing the superposition operator. As a drawing it is represented as a cloud (hence, the name "misty state" used for a superposition state). This, again, would be a great opportunity to watch (or rewatch) Terry's video [12] off the book website at [11]. In text we can use the following two representations corresponding to each one of the situations shown above: H(W) = [W, B] and H(B) = [W, -B]. The notation says, in essence, that there are two outcomes and each one is equally likely to be measured.

[2] We emphasize then that we cannot clone but we can teleport an unknown, arbitrary quantum state.

Here's how we represent these two transformations graphically:

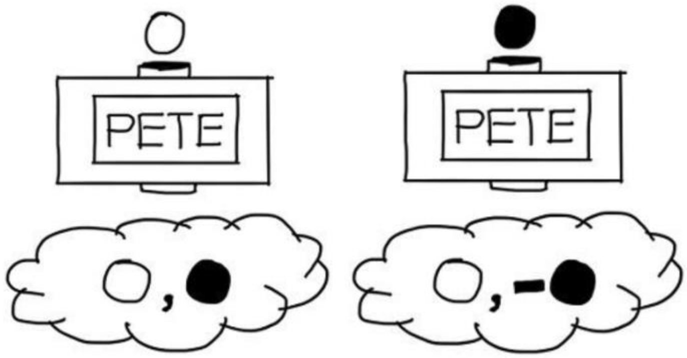


Figure 10. Misty states are superposition states. A negative sign (phase) shows on the right.

The Z Gate

The Z gate is introduced here as an exercise. Its definition is Z(W) = W and Z(B) = -B. Show that, just like for NOT, stacking two Z boxes leaves the input unchanged. We will soon learn that this is a general property of quantum gates and our next goal will be to prove it for PETE boxes (or Hadamard gates). That H(H(W)) = W and H(H(B)) = B is both non-trivial and very instructive. We can also demonstrate that experimentally in the Quantum Flytrap. In the end we would like to be able to show that this diagram commutes:

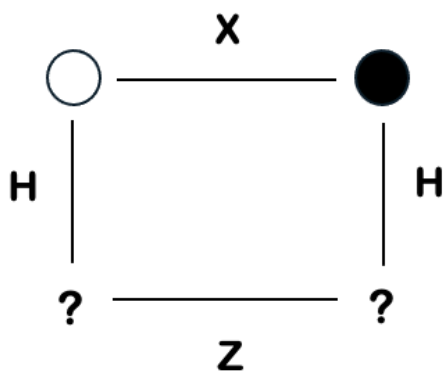


Figure 11. Exercise: complete this diagram and show that it commutes.

Linearity of Quantum Operators

We should first say that a misty state, so far, is in fact a sum of two states with probability amplitudes that are equal to each other. The phase we encountered thus far is simply a multiplication with the scalar -1. In quantum mechanics linearity of operators means that they satisfy two key properties: (a) they preserve the sum of states and (b) they preserve scalar multiplication. This property is fundamental to the superposition principle and the way quantum states evolve over time. Therefore we shall enforce it here.

As a result we have the following diagram showing how a NOT gate acts on a superposition of states:

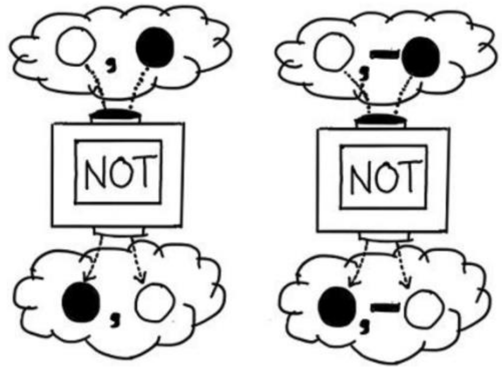


Figure 12. The effect of NOT gate on superposition of states.

We can describe what happens in Figure 12 as follows:

NOT([W, B]) = [NOT(W), NOT(B)] = [B, W] = [W, B].

We take the opportunity to point out here that like in a sum the order of factors (that is, the states in a superposition operator) does not matter so the NOT gate in effect leaves the first misty state unchanged. In the case of the second diagram we have:

NOT([W, -B]) = [NOT(W), NOT(-B)] = [B, -NOT(B)] = [B, -W] = -[-B, W] = - [W, -B]

We have in fact proved that these are the two eigenvectors of the NOT gate. In the process we illustrated linearity of phase and superposition operators with respect to the NOT gate. By a similar process we show how stacking two PETE boxes (or as everybody else knows them, Hadamard gates) leaves the input unchanged.

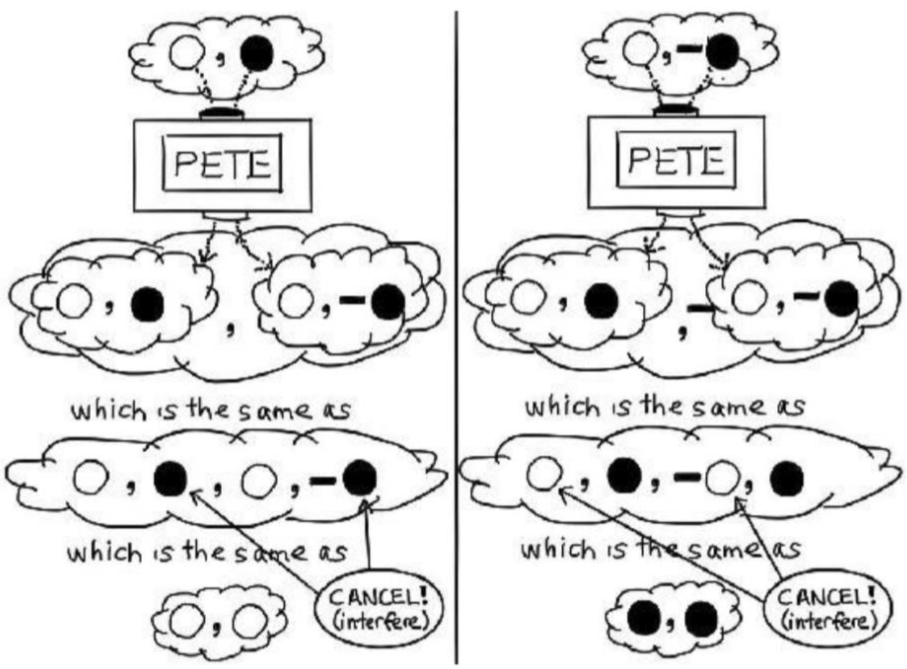


Figure 13. The effect of the PETE box (Hadamard gate) on two superpositions of states.

This part further uses the fact that a superposition operator is a sum and under certain conditions (i.e., when the superpositions are at the same depth and have the same number of distinct states) we can combine two mists by fading their boundaries so they can combine (join together) into a larger mist. Here's how this happens (in Fig. 13) in the notation we used to restate what was going on in Figure 12:

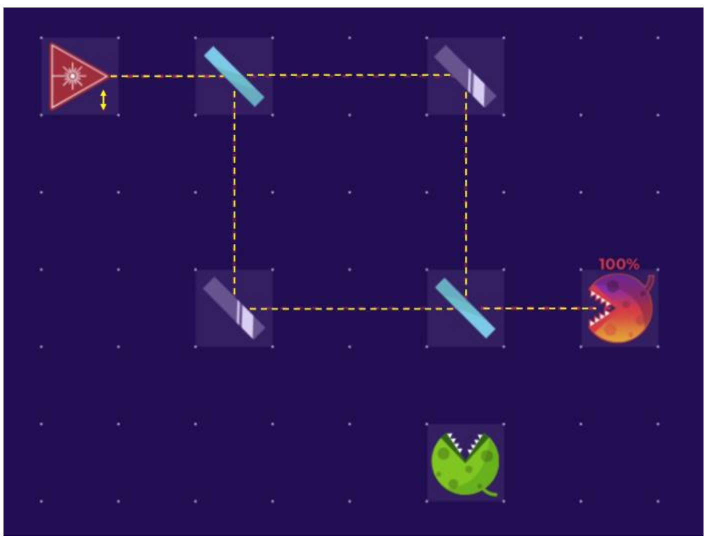
H(H(W)) = H([W, B]) = [H(W), H(B)] = [[W, B], [W, -B]] = [W, B, W, -B] = [W, W] = W

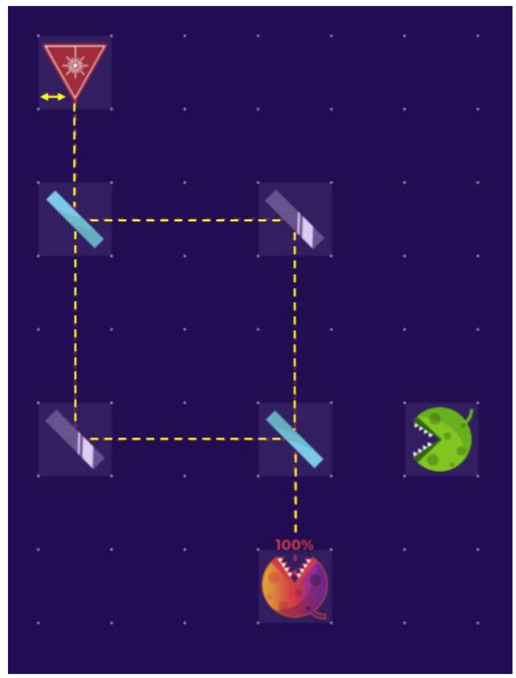
Please look at Figure 13 as it shows (diagramatically) what we wrote above, and below:

H(H(B)) = H([W, -B]) = [H(W), -H(B)] = [[W, B], [-W, B]] = [W, B, -W, B] = [B, B] = B

The next two figures show these relationships as diagrams, as seen in Quantum Flytrap.

Figure 14. Experimental verification/proof in Quantum Flytrap (via a Mach-Zehnder Interferometer) that stacking two PETE boxes leaves the input (vertical is B, horizontal is W) unchanged.





Systems of Two Qubits

There are four possible combinations of two qubits: WW, WB, BW and BB. We can represent this with white and black balls (or blobs) and we say that while they resemble multiplication they lack one important property of multiplication as they are not commutative. Thus WB and BW are different so order matters but other than that we can carry over some of the other properties encountered in multiplication: B [W, B] for example is the same as [BW, BB]. This, in effect, is how we define entanglement. Two (or more) particles are entangled when they are all described by the same wave function. For us this means that the expression that representes the state of the two (or more) qubits can't be separated as a product of factors each representing an individual qubit. Thus, because [BW, BB] = B [W, B] this equation does not describe a system of two entangled qubits. However a state like [BW, WB] cannot be split into a product of two states and thus represents an entangled state of two qubits (it's one of the Bell states). There is no entanglement in the Bernstein-Vazirani challenge that we discuss but we will be working with systems of two qubits and so we wanted to clarify this up front.

The phase kickback is the following situation (that is, this is what we need to prove):

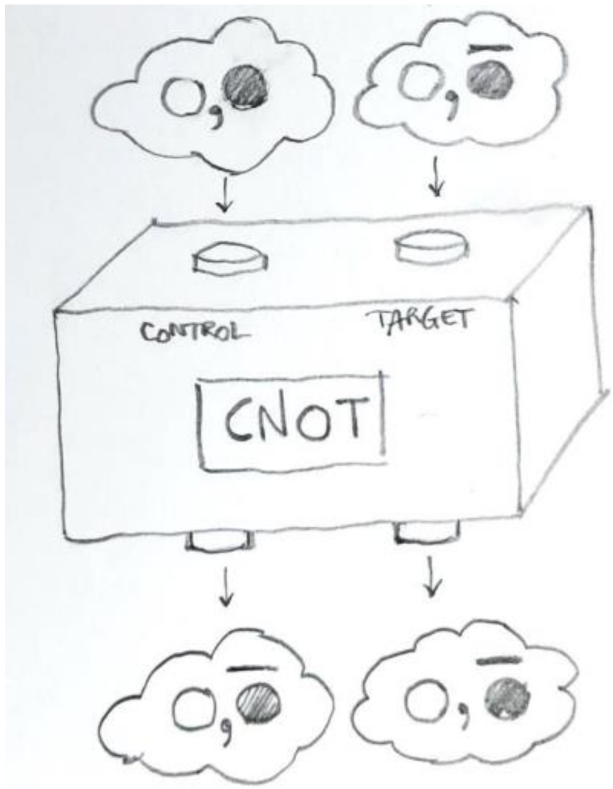


Figure 15. In quantum computing, operations have the ability to introduce phase changes to quantum states. When a controlled operation is applied to two qubits, the phase of the second (target) qubit is conditioned on the state of the first (control) qubit. Because here the phase of the second qubit is being “kicked back” to the first qubit, this phenomenon was coined “phase kickback” in 1997 by Richard Cleve, Artur Ekert, Chiara Macchiavello and Michele Mosca through a paper that solved the Deutsch-Josza problem.

How do we prove that that’s what happens above? Let’s start by writing the input as a system of two qubits. It is convenient here to keep the second qubit as a superposition and work with it as such. We have:

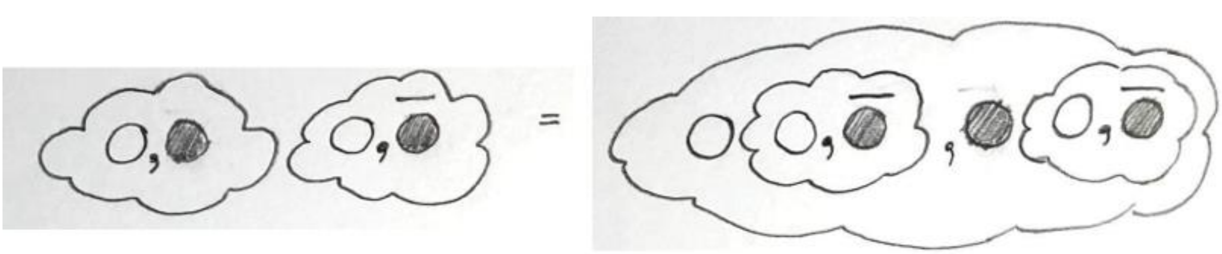


Figure 16. The input [W, B] [W, -B] = [W [W, -B] , B [W, -B]] in diagrammatic form.

Now that we have the input expressed as such let's pass it through the C-NOT gate and transform it using the rules of engagement already mentioned for this gate. We have:

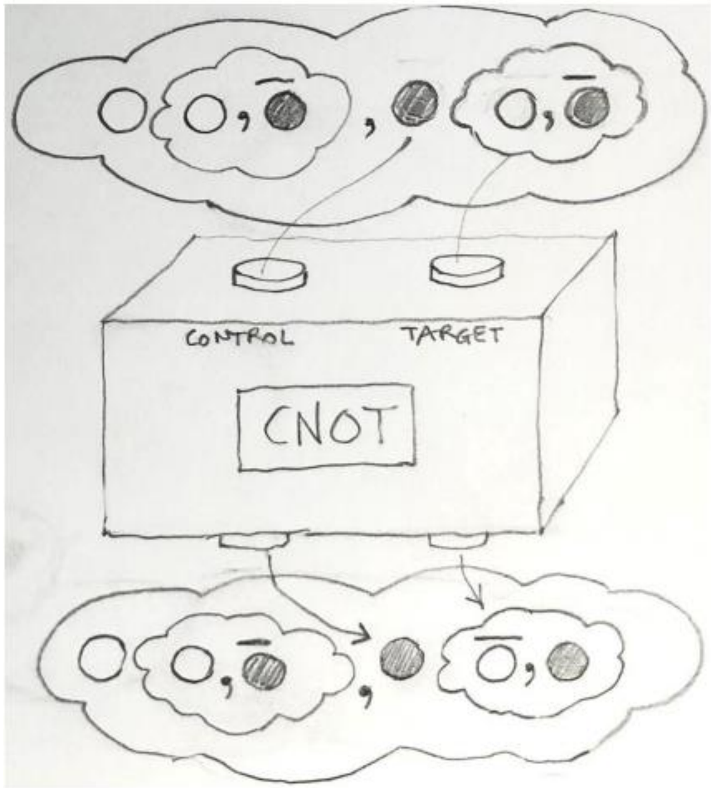


Figure 17. The effect of the C-NOT gate on our two qubit input. As shown in figure 17 each pair of qubits passes through the C-NOT gate. The first one is placing a W on the control which means the gate will leave the second qubit unchanged. The second pair has a B on the control which means that the second qubit is flipped. The rule for flipping a superposition of states (via NOT) has been shown before and it's like in Figure 18.

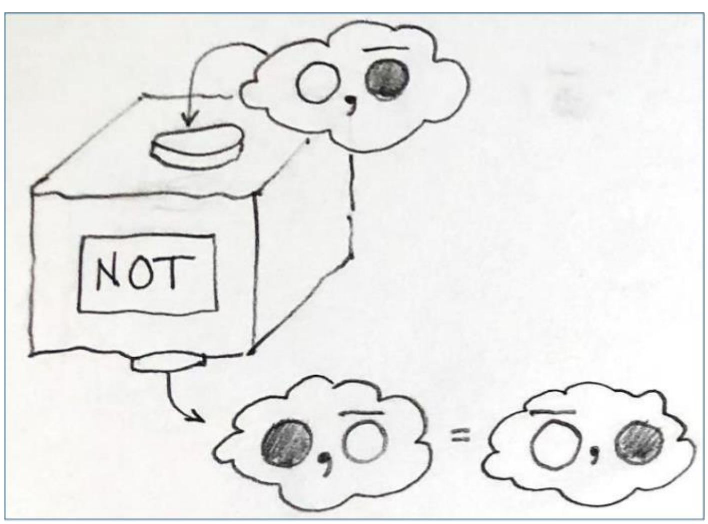


Figure 18. The effect of the NOT gate on a superposition of states.

The purpose of Fig. 18 is to support the transformation shown in Fig. 17. Thus, some readers might consider the picture to be be redundant while some might prefer to use shorthand to describe it, e.g. NOT([W, -B]) = [-W, B]. We've gone over this earlier when we said that the input here is one of the two eigenvectors of the NOT gate. Since the superposition operator is actually a sum (as we said before) the order of states in a mist is not important but an order is usually preferred and the phase distributes over the constituent states, as shown in the picture below:

Figure 19. A negative phase applied to a mist distributes over its constituent states.

Now we can rewrite the second state in the output of Figure 17 as follows:

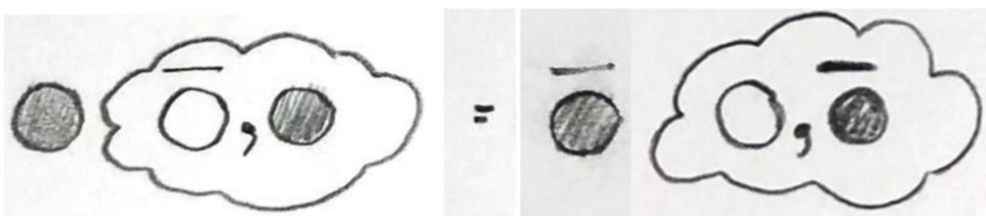


Figure 20. Moving the sign (phase) from the second qubit to the first has this effect.

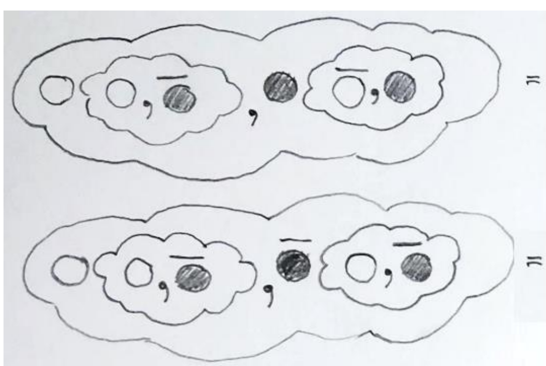
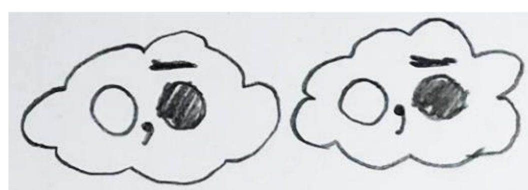


Figure 21. The output state from earlier (Fig. 17) is shown on the left. It can be reformulated as mentioned and rewritten (reverse FOIL method) as a product of states, each one a superposition. We're now finished.



So now we have proved the following:

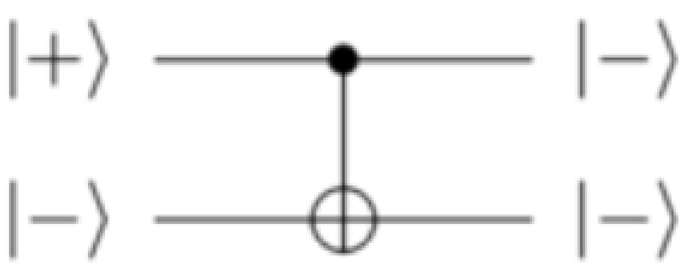


Figure 22. Phase kickback, conventional notation.

We're now ready to solve the Bernstein-Vazirani challenge.

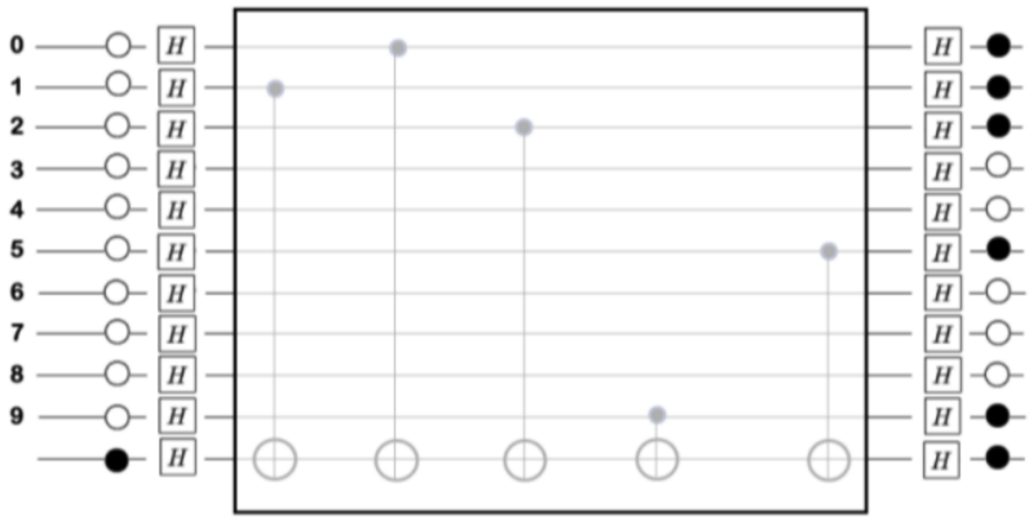


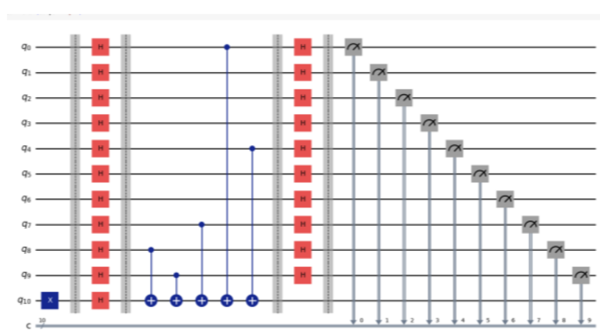
Figure 23. Bernstein-Vazirani challenge: the solution. If the C-NOT gates inside the black box had been oriented the other way the solution would have been immediate. But since they're as shown above one would need to test every input in part, thus establishing a linear lower bound for the complexity of finding the pattern. With the help of the previous result and a corresponding number of Hadamard (PETE) boxes we can determine the structure of the black box in one step. Note that the order of the gates is irrelevant so the musical score that we included in the initial diagram does not really apply (since for the music the order of the notes does indeed matter) but it gives us an opportunity to make this point here. Below we show how this challenge can be implemented in Qiskit:

[1]
```
!pip install qiskit qiskit-aer
```

[2]
```
import qiskit
```

[3]
```
from qiskit import *
from qiskit_aer import AerSimulator
```

[4]
```
secretnumber = '1110010001'
indices = [1, 0, 2, 9, 5]
n = len(secretnumber)
circuit = QuantumCircuit(n+1, n)
circuit.x(n)
circuit.barrier()
circuit.h(range(n))
circuit.h(n)
circuit.barrier()
for index in indices:
  circuit.cx(n - index - 1, n)
circuit.barrier()
circuit.h(range(n))
circuit.barrier()
circuit.measure(range(n), range(n))
circuit.draw(output='mpl')
```



[5]
```
from qiskit_aer import AerSimulator
device = AerSimulator()
shots = 1
job = device.run(circuit, shots = shots)
result = job.result()
counts = result.get_counts(circuit)
```

[6]
```
print(counts)
```

```
{'1110010001': 1}
```

Figure 24. Creating and measuring the quantum circuit for the Bernstein-Vazirani challenge

Note that the order of qubits is reversed in Qiskit so the circuit is reflected: as an example, the third C-NOT in the black box in our drawing connects the bottom line (its target) to the third line from the top (the control). In the Qiskit circuit it connects the bottom line (target) with the third line from the bottom (Qiskit numbers the qubit lines in reverse order). Also, as we mentioned earlier, the order of the gates in the black box is not important; the order of the lines in the input and output is (and the output is determined, as predicted, in one shot).

Irreducible Misty States

In the original (pure) misty state formalism two or more separate mists can be combined when (and only when) they (a) are at the same level and (b) have the same cardinality. This leads us to states that are irreducible. For example, see Figure 25, below:

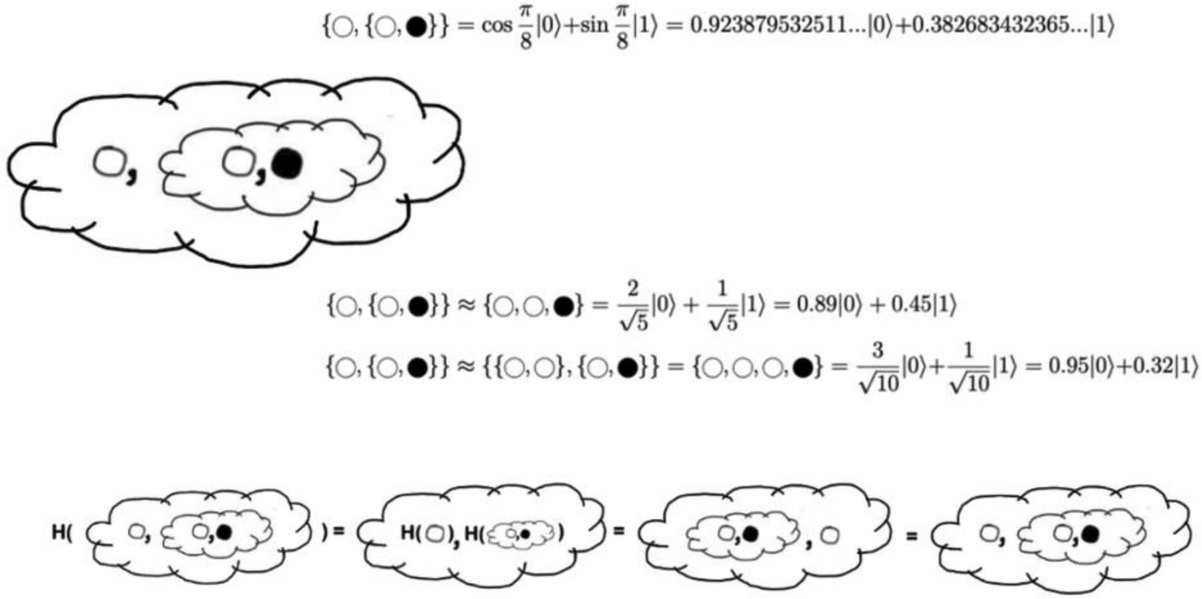


Figure 25. Irreducible misty states; the one presented here is a Hadamard eigenvector.

The figure above shows three things: (a) there is a straightforward conversion from any misty state to an algebraic expression; (b) that the original, irreducible state can be approximated reasonably well within the original (pure) misty state formalism and (c) that the state we have chosen (smallest irreducible state with mist inside mist) is in fact one of the eigenvectors of the Hadamard gate. As an aside, but an important one, this eigenvector of the Hadamard gate gives us an alternative path to extending the original (pure) misty state formalism. We only need ask the question[3]: "Is there any mist which passes through the Hadamard gate such that the probabilities of observing a white ball or a black ball are unchanged?". And if we write the equations and solve them we find that the ratio of white to black balls in that state has to be an irrational number. (And now we can also tell the story of poor Hippasus.)

[3] Due to Terry Rudolph (personal communication).

Abstracting and Representing Non-Classicality

In this section we want to show how misty states lead naturally to situations illustrating minimal examples of non-classicality. As an example consider the following well-formed (but irreducible) misty state: $\{🍎, \{🍉, 🍒\}\}$. It says that three types of fruit are possible for a snack: apple, watermelon and cherries. The specific item that we will end up with is determined probabilistically (via measurement). We do know it will be one of the three items listed. In this misty state the watermelon and cherries are in equal superposition with each other (let's call that state s_1) and the apple is in equal superposition with the state s_1. When we estimate the probability of each outcome we find that the probability of receiving an apple comes up as $p(🍎) = \frac{1}{2}$.

Meanwhile for the other two items $p(🍉) = p(🍒) = \frac{1}{4}$ (so each has a probability of 0.25).

Now if we modify the misty state to $\{🍎, \{🍎, 🍒\}\}$ we have a minimal non-classical situation. In this new state one would expect the probability of getting an apple to still be 0.75, when in effect it becomes[4] 0.853 (thus lowering the chance of receiving cherries to 0.147).

The resulting state is (in standard mathematical notation):

$$|\Psi\rangle = \frac{\sqrt{2+\sqrt{2}}}{2}|🍎\rangle + \frac{\sqrt{2-\sqrt{2}}}{2}|🍒\rangle = \cos\frac{\pi}{8}|🍎\rangle + \sin\frac{\pi}{8}|🍒\rangle$$

Of course, the misty state representation of this situation is far simpler (although it properly belongs to the extended misty state formalism). Figure 26 and 27 present the same story in Quantum Flytrap: the vertical beam represents cherries, the horizontal beam stands for apple. The probabilities of observing each one of these orthogonal states (with interference, as shown in Figure 26) are as predicted (i.e., the minimal non-classical situation we have identified).

If we block the interference and (as shown in Figure 27 measure each path independently) we find the evidence that when the two arms of the MZI are separated (and we know exactly down which path each photon traveled) our measurements will lead to the classical result. But once they combine paths, the probabilities change. This is the indistiguishability principle in action: when a photon can take more than one path to the detector and the detector can't determine which actual path the photon has taken we have interference. In Figure 26 there are two ways to get an apple but when we do we can't tell if it's due to the apple in the outer or the inner mist.

[4] The reason, of course, is that the probability amplitudes add up, but probabilities do not.

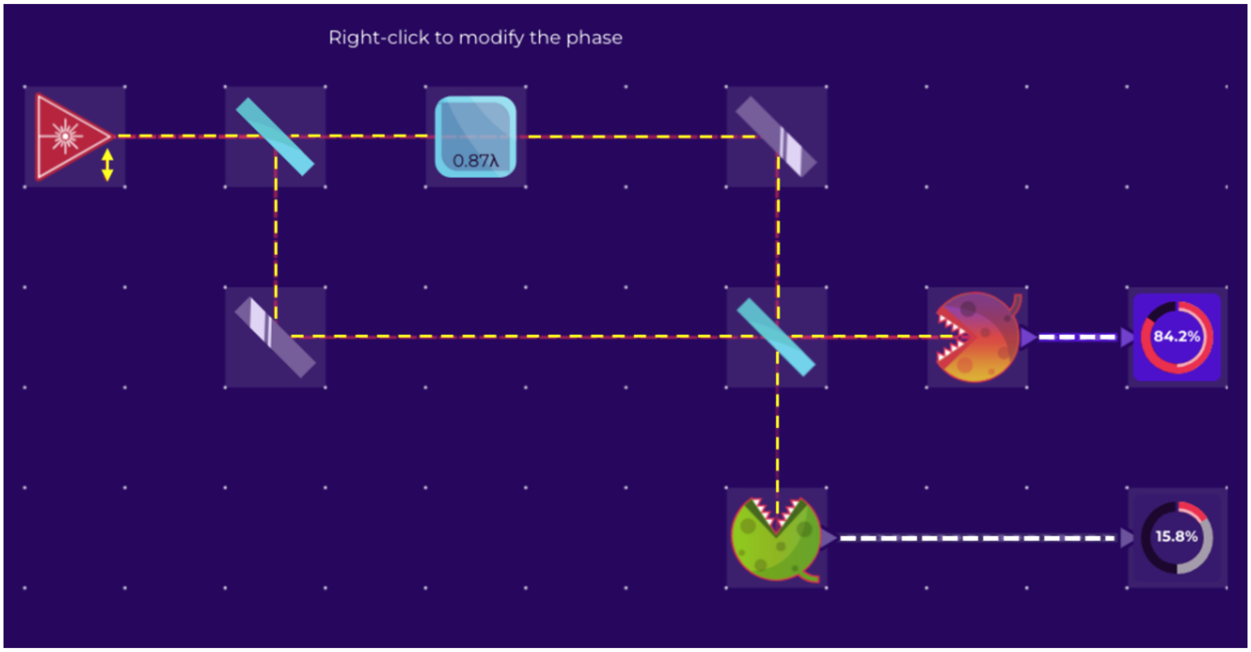


Figure 26. For this quantum snack we first create a superposition (state s_1) via rotation with 45 degrees to control the phase difference between the two arms of the Mach-Zehnder Interferometer (MZI). We then add the unrotated vector to obtain the final result. Similar experiments are discussed in Scarani's papers on one-particle quantum interference.

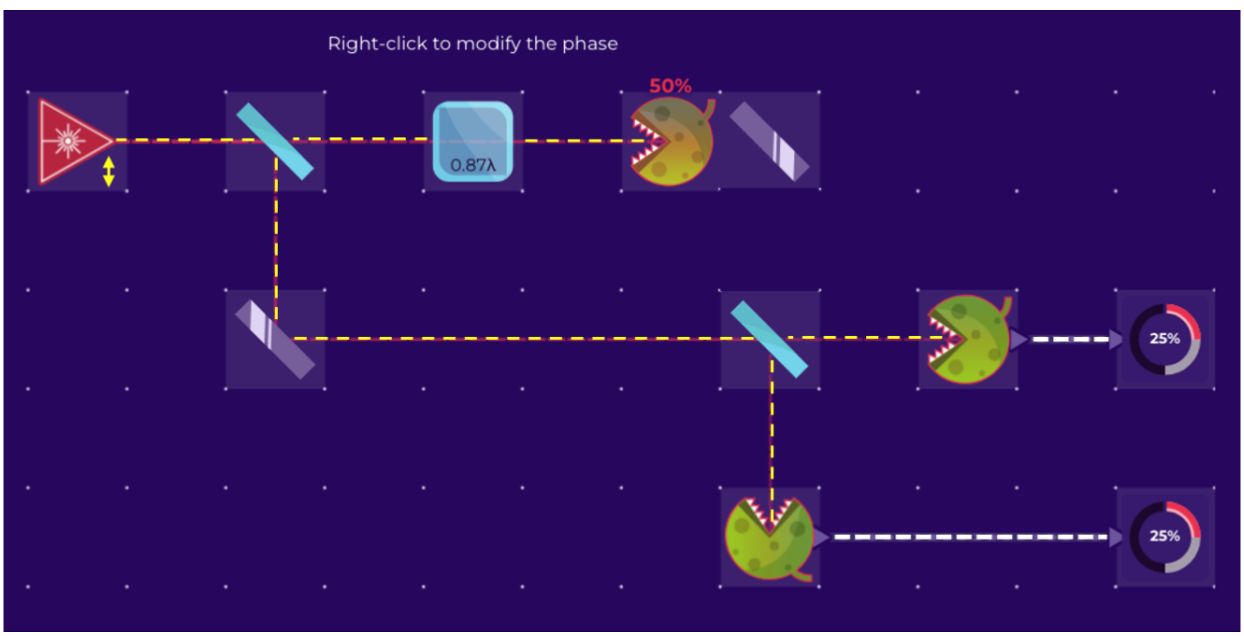
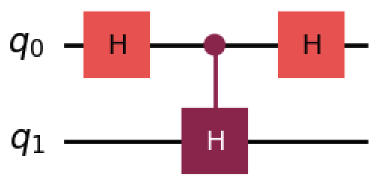


Figure 27. Before interference we measure the same values for probabilities as in the classical case. If we remove the detector in the top arm of the MZI we obtain the situation from Figure 26.

The next section ends the paper and presents our conclusions. Before we do that we would like to share another exercise with the reader. Consider the following circuit:

```
[3] from qiskit import QuantumCircuit
```

```
[5] qc = QuantumCircuit(2)
    qc.h(0)
    qc.ch(0, 1)
    qc.h(0)
    qc.draw(output='mpl')
```




```
from qiskit.quantum_info import Statevector
a = Statevector.from_instruction(qc)
a.draw('latex')
```

$$0.8535533906|00\rangle + 0.1464466094|01\rangle + \frac{\sqrt{2}}{4}|10\rangle - \frac{\sqrt{2}}{4}|11\rangle$$

Figure 28. Show that this circuit produces both eigenvectors of the Hadamard gate.

Exercise: Prove, using the pure misty states formalism, that the circuit in Fig. 28 produces both eigenvectors of the Hadamard gate; also, please explain how it achieves that goal. What is a simplest such circuit? Can you think of something simpler than what we have above? Is there a one-qubit circuit that achieves the goal of producing one, or the other, eigenvector of the Hadamard gate (also known as the PETE box)? A two-qubit circuit? How does that work?

Conclusions and Acknowledgments

CS2023 makes some excellent recommendations (for the first time ever) on how to include a knowledge unit on quantum information science, computing and quantum algorithms. Their proposal is organized in three stages and comprises a short (eight-weeks) class, a one semester class and a longer, two semester sequence that (at least in principle) makes heavy use of a lab (or fab, depending on resources) in quantum hardware, gates and circuits.

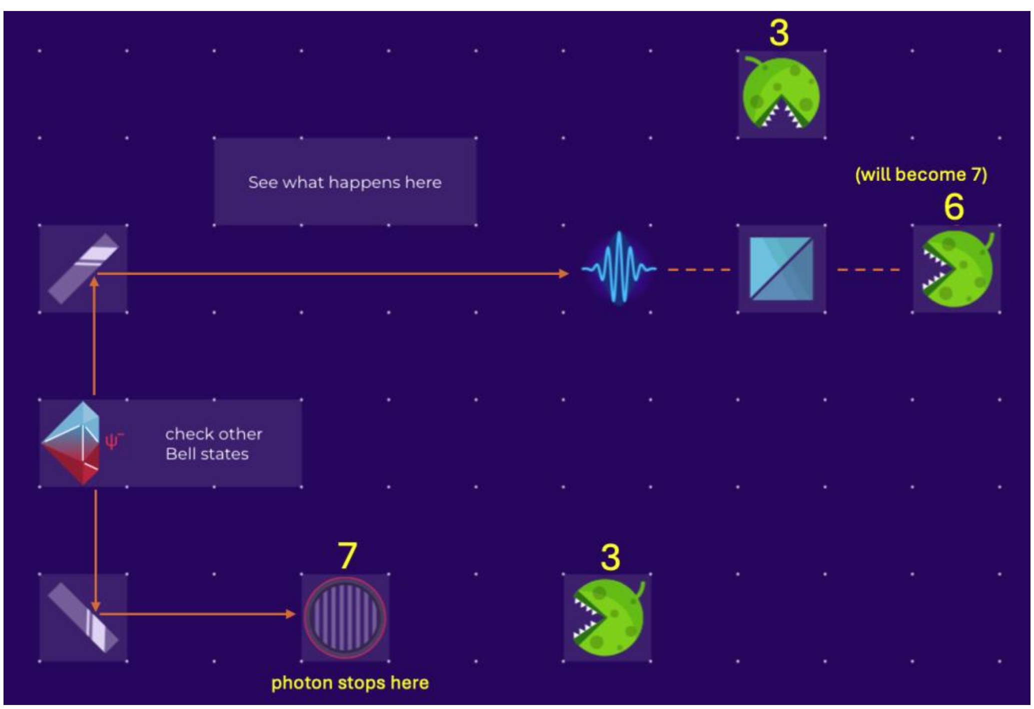
Following those recommendations we have described here our approach of implementing the eight-week syllabus with extended material from Terry Rudolph's groundbreaking "Q is for Quantum". This material has been tested in the classroom, in various conferences in workshops and tutorials, and at many levels – including creating a faculty learning community (FLC) for HS and middle school CSCI teachers in the state last summer with significant support from the Computer Science Teachers' Association (CSTA) in our state.

Quantum Mysteries

Quantum superposition is a fundamental principle/mystery in quantum mechanics (and at the heart of the solution to the Bernstein-Vazirani challenge that was presented in this paper). It states that, much like waves in classical physics, any two (or more) quantum states can be added together ("superposed") and the result will be another valid quantum state; and conversely, that every quantum state can be represented as a sum of two or more other distinct states. Mathematically, it refers to a property of solutions to the Schrödinger equation; since the Schrödinger equation is linear, any linear combination of solutions will also be a solution.

The other fundamental "mystery" in quantum mechanics is entanglement. An entangled system is defined to be one whose quantum state cannot be factored as a product of states of its local constituents, that is to say, they are not individual particles but are an inseparable whole—if entangled, one constituent cannot be fully described without considering the other(s). The state of a composite system is always expressible as a sum, or superposition, of products of states of local constituents; thus, it is entangled if this sum necessarily has more than one term. Entanglement is a subtle concept and students exposure to it needs to be planned with care (see below). For emphasis, in a system of two entangled qubits the quantum system seems to acquire a probability distribution for the outcome of a measurement of the second qubit upon measurement of the first qubit in such a way that this probability distribution is different from what it would have been without the measurement of the first particle. Two particles whose future is described by one single wave function: this may definitely be perceived as quite surprising in the case of spatially separated particles!

Appendix



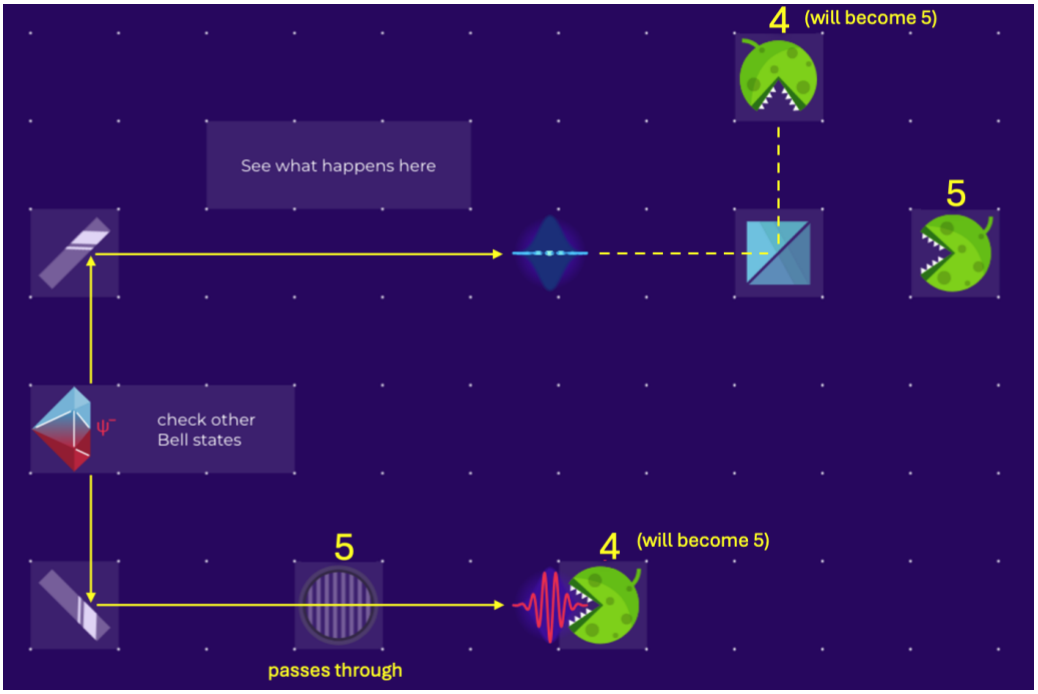


Figure 29. Entanglement is capable of instantaneous (i.e., superluminal) correlations even though information still can't travel faster than light (a classical channel is needed). Another no-go theorem states that an arbitrary quantum state can't be cloned (and the no-broadcast theorem generalizes no-cloning to mixed states). Quantum teleportation is then used to move a quantum state from one location to another (and in the process the state is consumed at the source). Entanglement is an important part of that protocol and our hope[5] that one day quantum repeaters will become reality. Quantum Flytrap is a free online environment that provides a wealth of animated and interactive quantum experiments; above we show two screen shots from the "spooky action at a distance" module/experiment.

We could also include here the solutions to the two challenges issued to the reader, for example the last one:

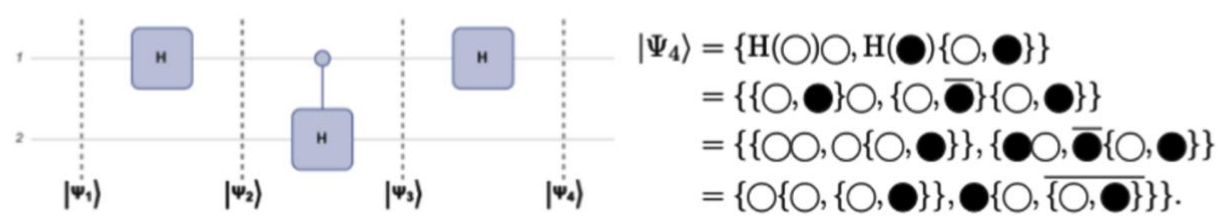


Figure 30. Proof that the circuit produces both eigenvectors of the Hadamard gate, alternatively. If one measures the first qubit and obtains W (that is, a 0 (zero)) the second line has the first eigenvector of the Hadamard gate. If the measurement on the first qubit line yields a 1 (that is, a one, indicated by B) the state of the second line resolves into the second eigenvector of the Hadamard gate. Note calculation is done entirely in the pure misty state formalism from

[5] Teleportation of entanglement is called entanglement swapping.

the book (in spite of the fact that the representation of the two eigenvectors is irreducible in that formalism). That means that the formalism in the book, as simple as it may appear to be, is not only surprisingly effective over a large number of well-known examples that are part of any serious introduction to quantum computing and quantum information science but at the same times gracefully transcends the examples in the book. We are now in the process of using the misty state formalism to describe the ZX calculus (transformations and results) described in[6] Quantum in Pictures. These two approaches are orthogonal, complementary not competing and/or conflicting and together are at this time the most successful methods used to introduce quantum to teenagers in Europe (UK), US and elsewhere.

References

1. Burd, B. *Quantum Computing Algorithms: Discover how a little math goes a long way.* Packt Publishing, Sep 22, 2023. First edition.

2. Ekert, A. (adviser) Quantum Flytrap: A No-Code IDE for Quantum Computing. https://quantumflytrap.com/ Accessed: 2025 May 21.

3. Ellwanger, D. *A Quantum Computing Pamphlet*. Independently published, Sep 8, 2019.

4. Fayer, M.D. *Absolutely Small: How Quantum Theory Explains Our Everyday World.* Amacom Books, Special Edition, June 16, 2020.

5.

6. Gigerenzer G. Embodied Heuristics. Frontiers in Psychology 2021 Sep 30 https://tinyurl.com/embodied-heuristics URL accessed successfully on 2025 May 21

7. Haldane, A.G. and Madouros, V. The Dog and the Frisbee. Presented at the Fed. Reserve Bank of Kansas City's annual economic policy symposium Aug 30-Sep 1, 2012 in Jackson Hole, Wyoming. https://tinyurl.com/frb-of-kc, has link to the talk/presentation/materials: https://tinyurl.com/the-dog-and-the-frisbee both accessed: 2025 May 21.

8. Marinescu, D.C. and Marinescu, G.M. *Approaching Quantum Computing*. Pearson Education Inc., Sep 20, 2004.

9. Nielsen, M.A. and Chuang, I.L. Quantum Computation and Quantum Information. Cambridge University Press, January 31, 2011. Tenth anniversary edition.

10. Rudolph, T. *Q is for Quantum*. Published by Terence Rudolph, July 19, 2017.

11. Rudolph, T. Book website: https://www.qisforquantum.org accessed: 2025 May 21.

12. Rudolph, T. https://tinyurl.com/terry-rudolph-misty-states accessed: 2025 May 21

13. Watrous, J. Understanding Quantum Information and Computation (IBM Quantum Education Series of Video Lectures) https://tinyurl.com/watrous successfully accessed on 2025 May 21.

---

[6] https://quantuminpictures.org/resources/