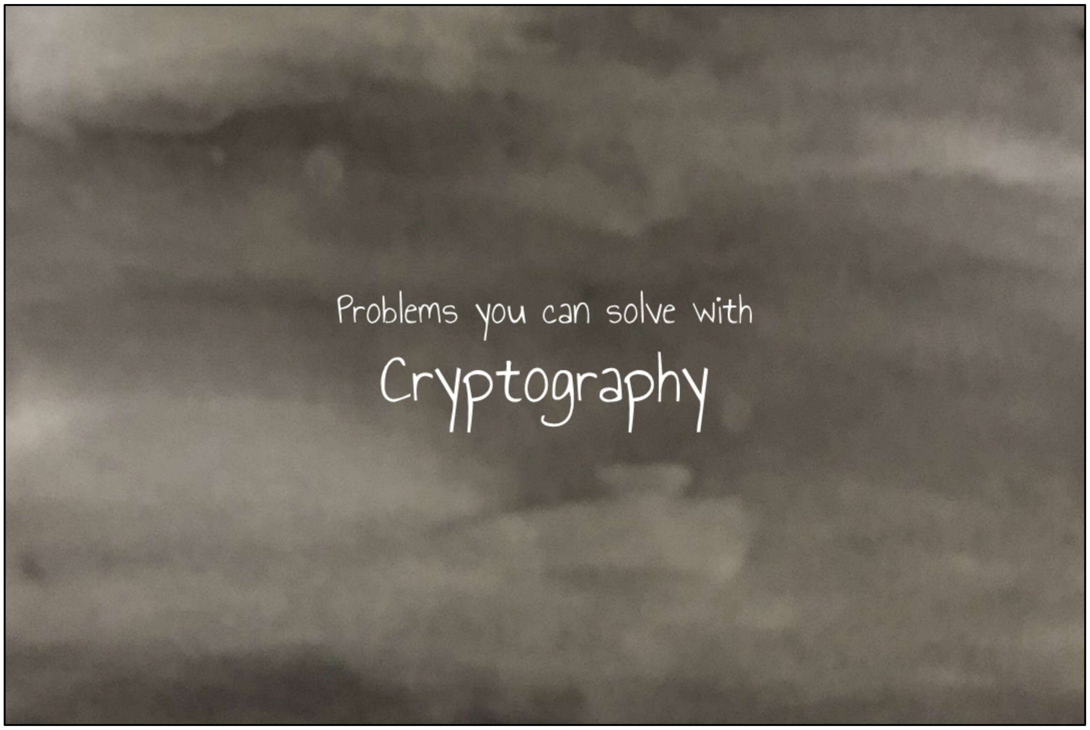


Problems you can solve with

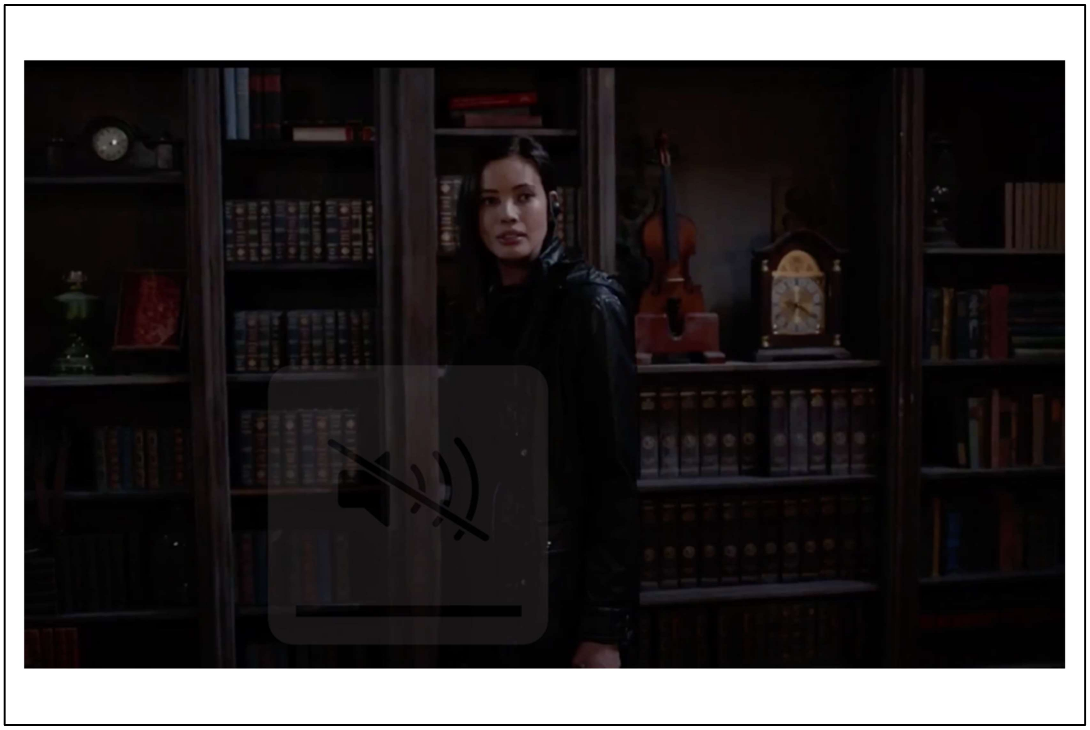
Cryptography

So, I just wanted to talk about two (unrelated) interesting topics in cryptography.

In particular, we're going to work through two problems I find quite interesting, and then discuss the cryptographic methods that we use to solve them.

First up....

The first question....



...has to do with this (cheesy and also amazing) show I've been watching called Deception.

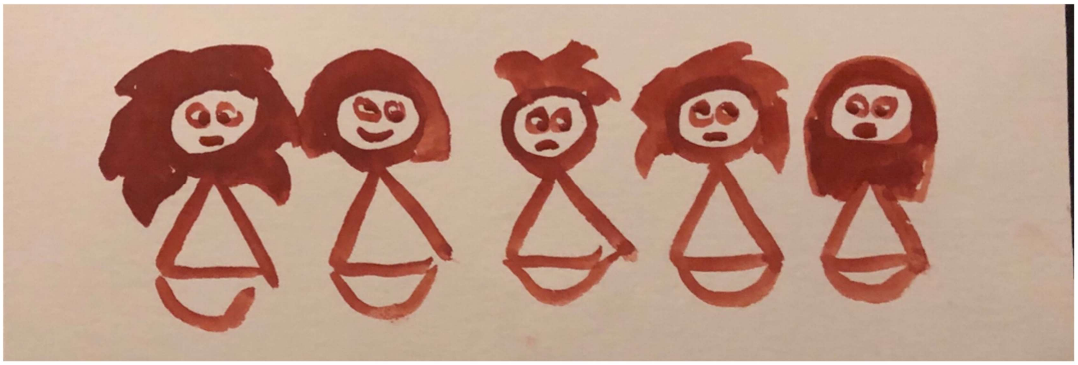
I'm going to kind of spoil the end of season 1. One of the characters wanted to share the contents of a secret vault with his five different billionaire friends, but didn't want any one person to access the vault, because then they might steal the others' stuff. The way he decided to lock the vault, then, was that all five people who had access to the vault had to put various objects (a violin, a watches, a jewel, an ancient book, and a fancy plate) into the correct places in a room, and only then did the vault open. (Full clip of this scene: https://tinyurl.com/deception-vault-scene).

Now say you had a vault that you wanted only to be cracked open when everyone in a particular group of your billionaire friends was there. It turns out it is possible to do — with no violins or special jewels required.

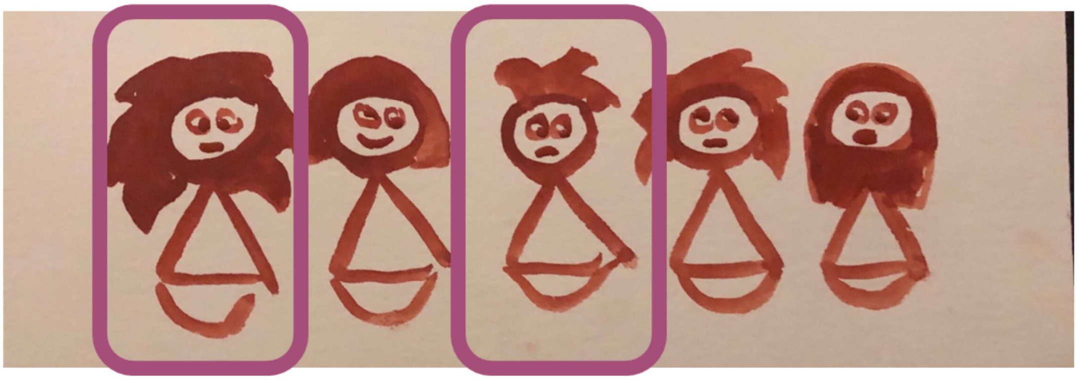
How could you do it?

One way is to build the vault with a keycode, and give each billionaire a digit of the keycode. One person gets 3, another gets 2, another gets 1, and then once they're all together, they can unlock the safe with the passcode 321.

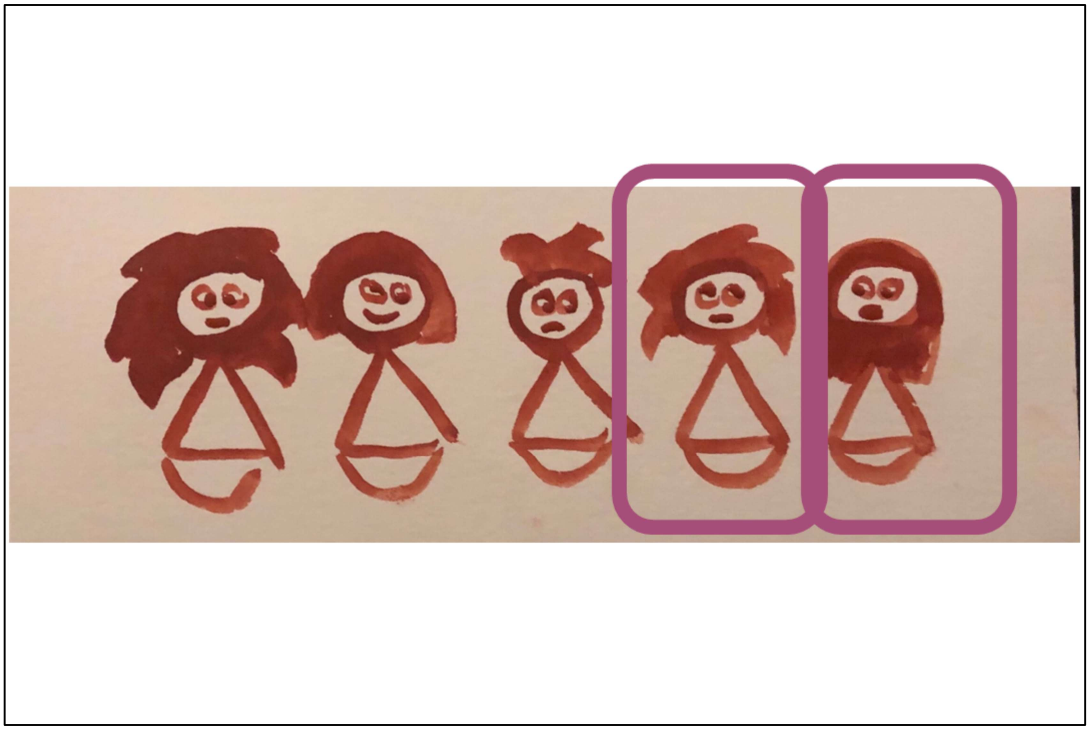
But now say your requirements are more complicated. Say you have five billionaire friends, but you want any two of the billionaires to be able to unlock the vault together (say, in case three of these billionaires die.)



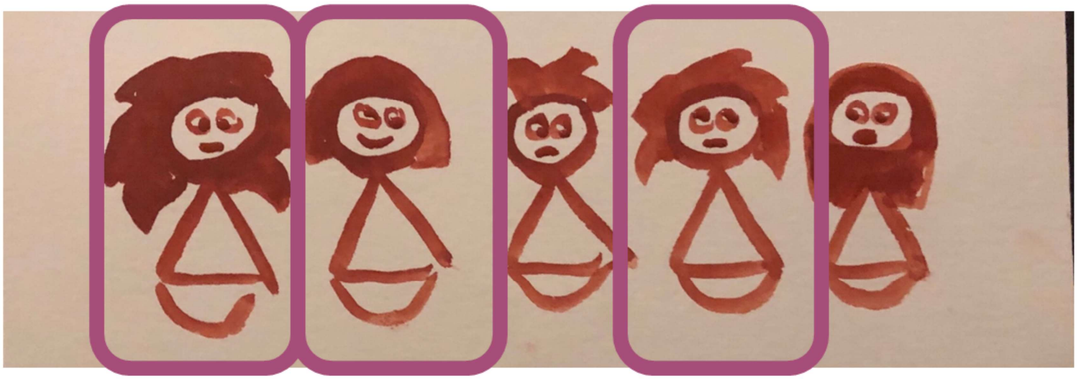
That is, you don’t need all five billionaires to be there to unlock the safe.



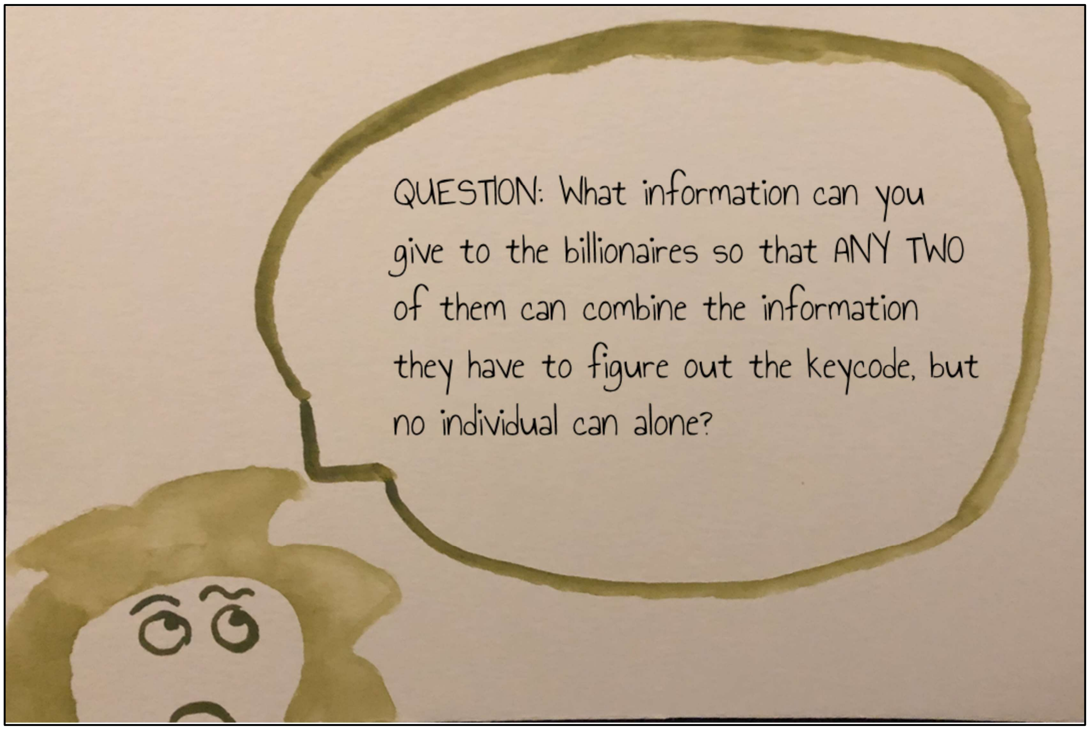
You’re ok if these two go to the vault together



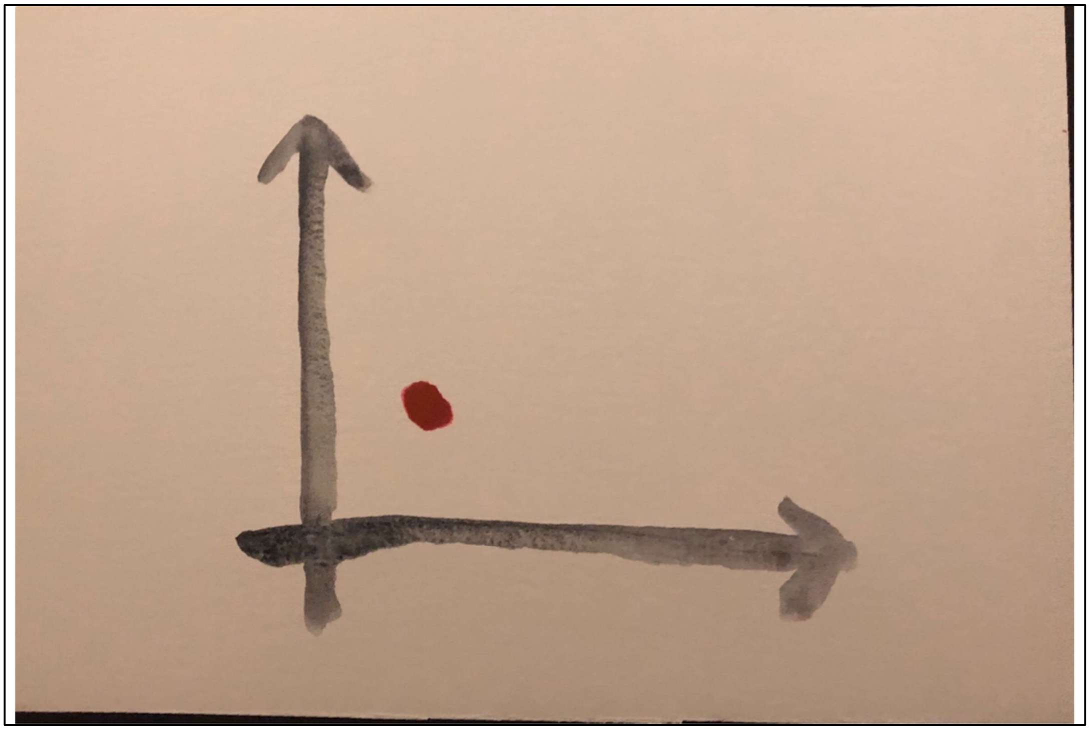
Or if these two go to the vault together.



Or if these three go to the vault together.

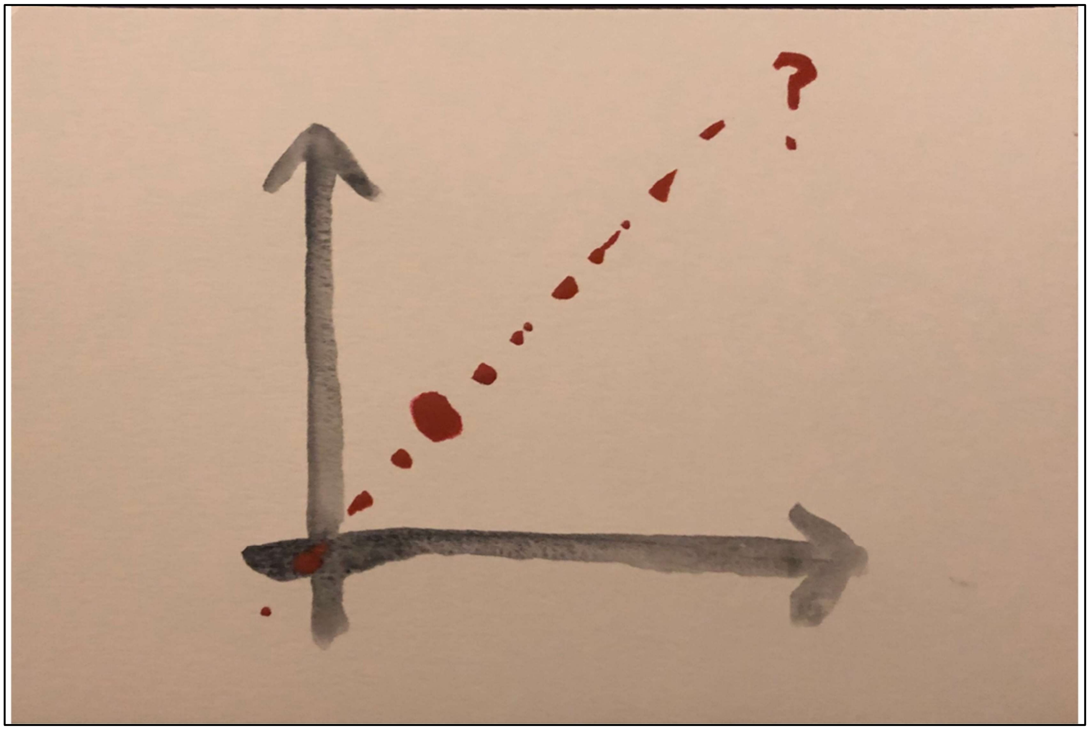

QUESTION: What information can you give to the billionaires so that ANY TWO of them can combine the information they have to figure out the keycode, but no individual can alone?

You just want at least two of the people to be there before the vault opens. How can you make it so that if ANY two of them put their information together, they'll come up with the keycode (but no individual can alone)?

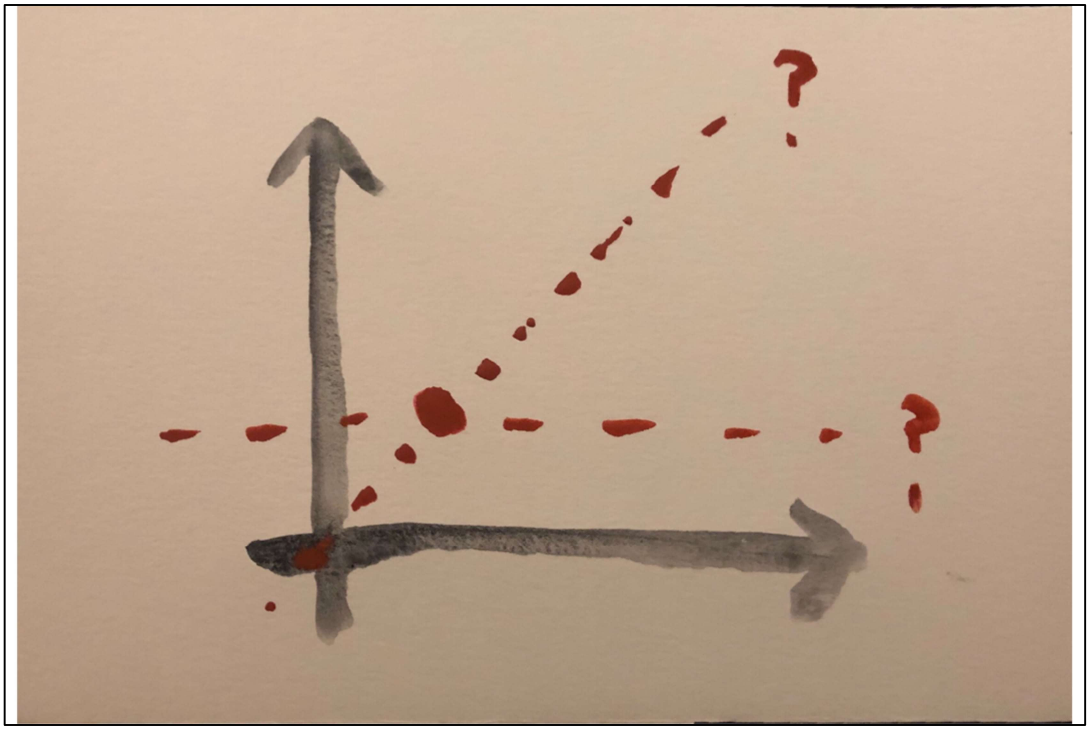


If you’re stuck, we can pause the thinking on that, and think of a related problem.

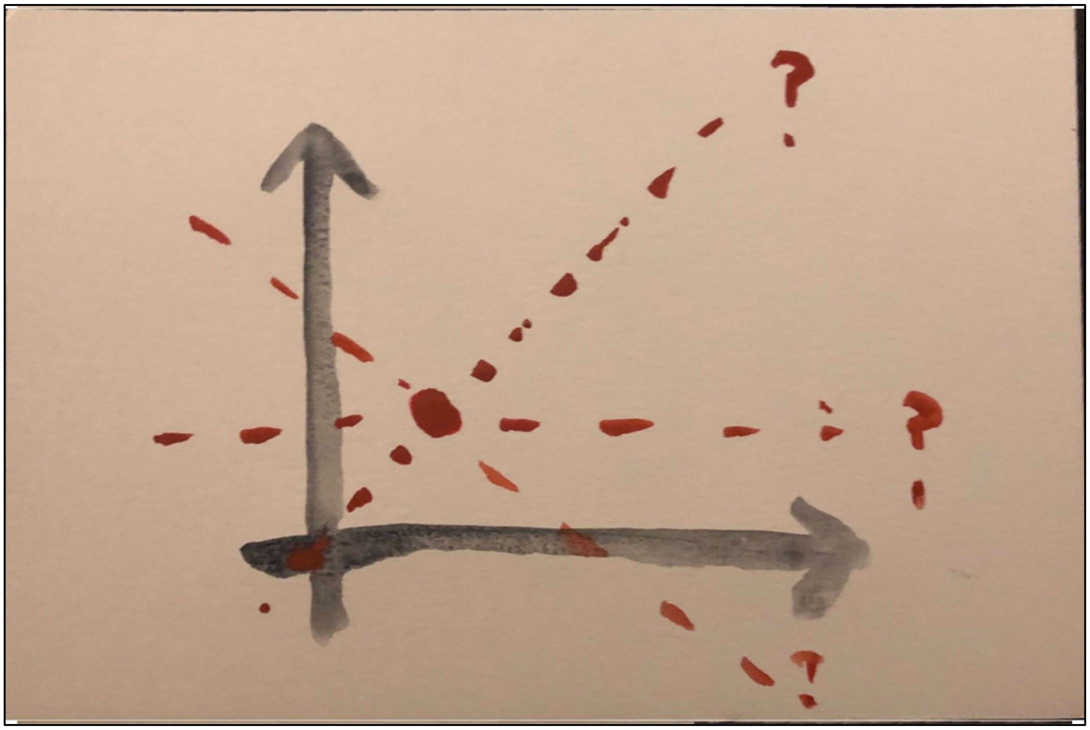
Let’s say I’m thinking of a line — a particular line on a plane. And let’s say I only tell you one point on my line. Then, there’s almost no hope that you’ll be able to guess my line.



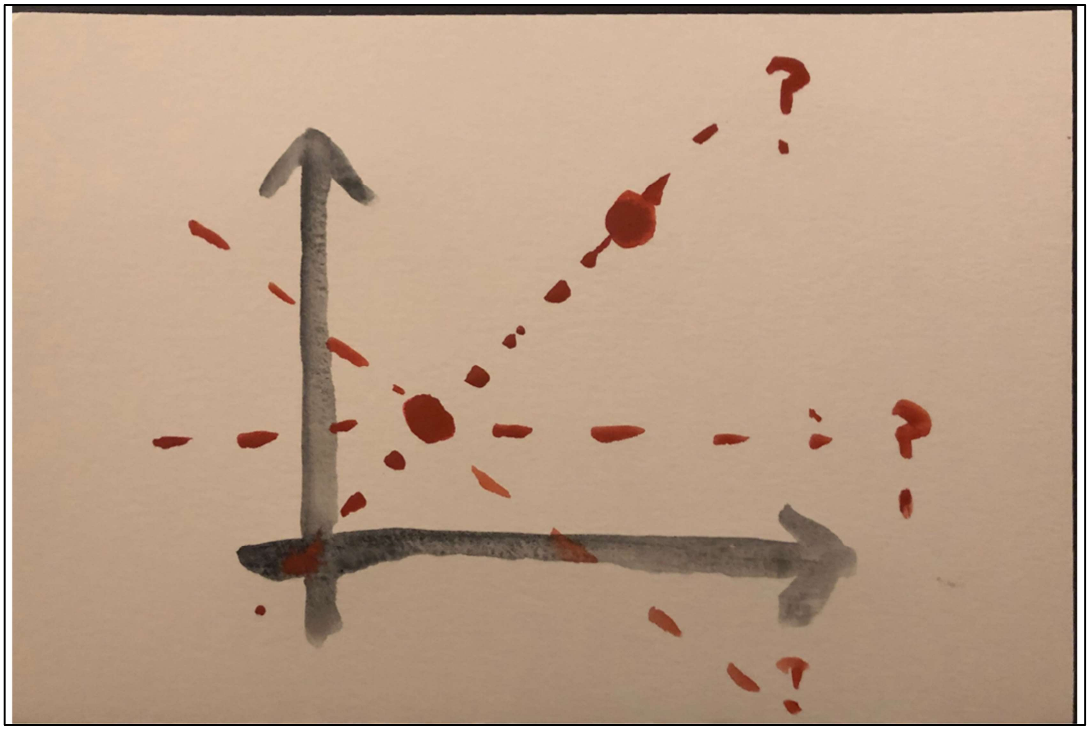
It could be this line



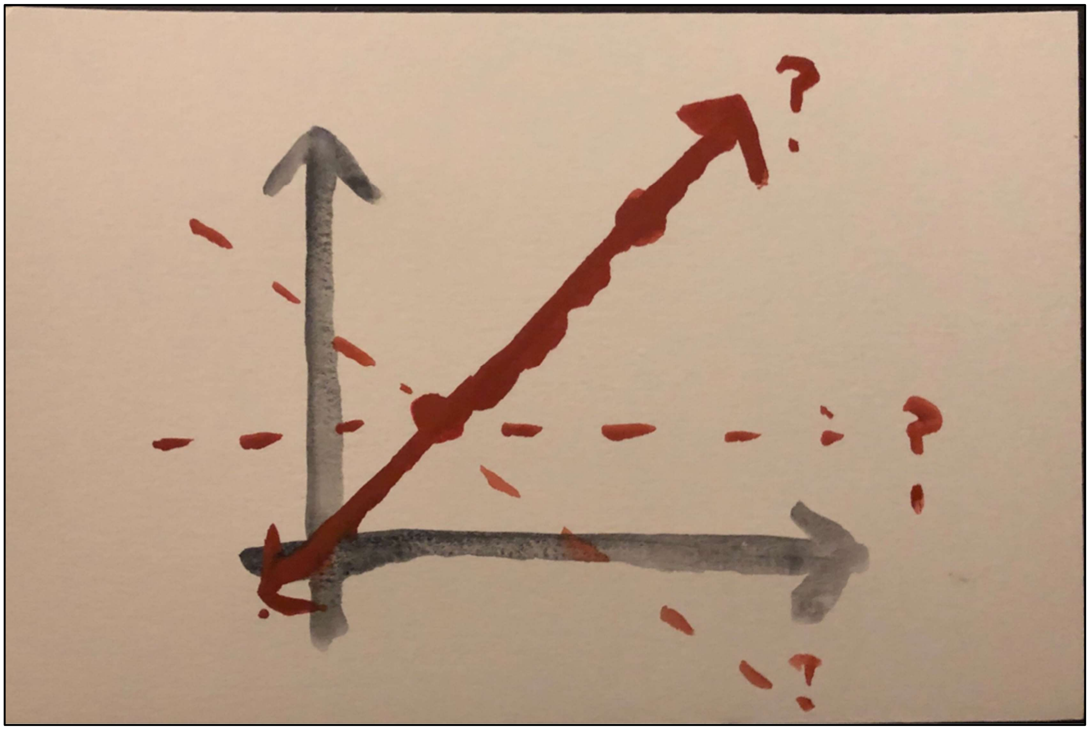
Or this one.



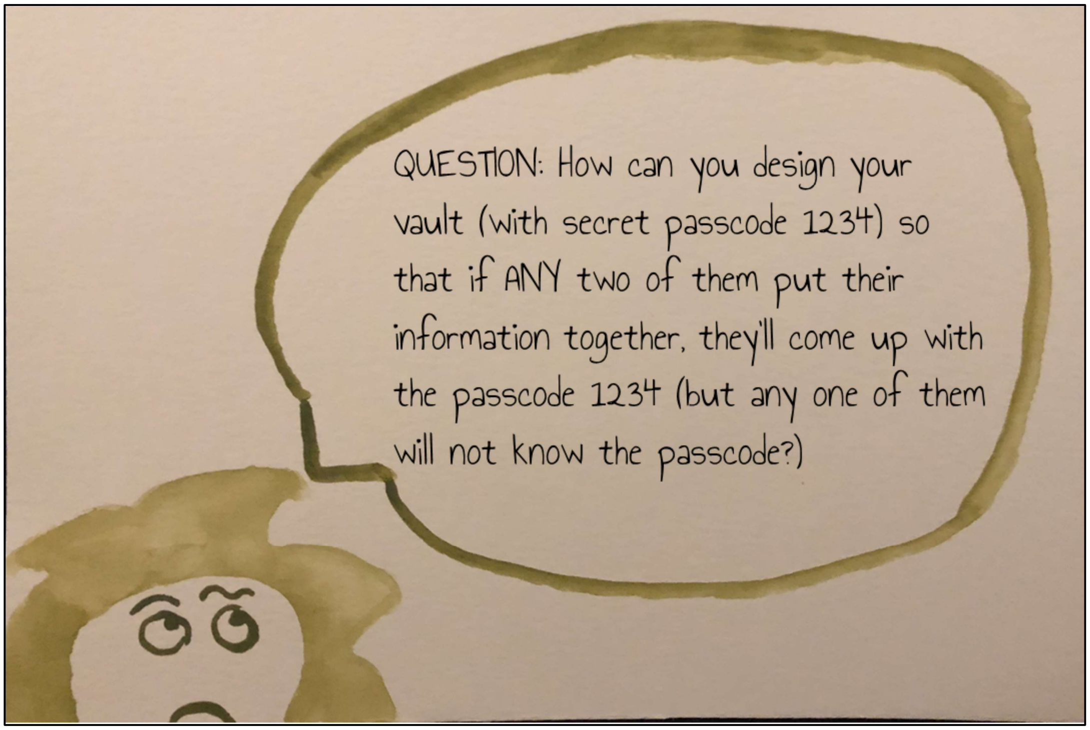
Or this one. Or any other infinite number of lines.



But let’s say somehow, you were able to find out about another point on the line.

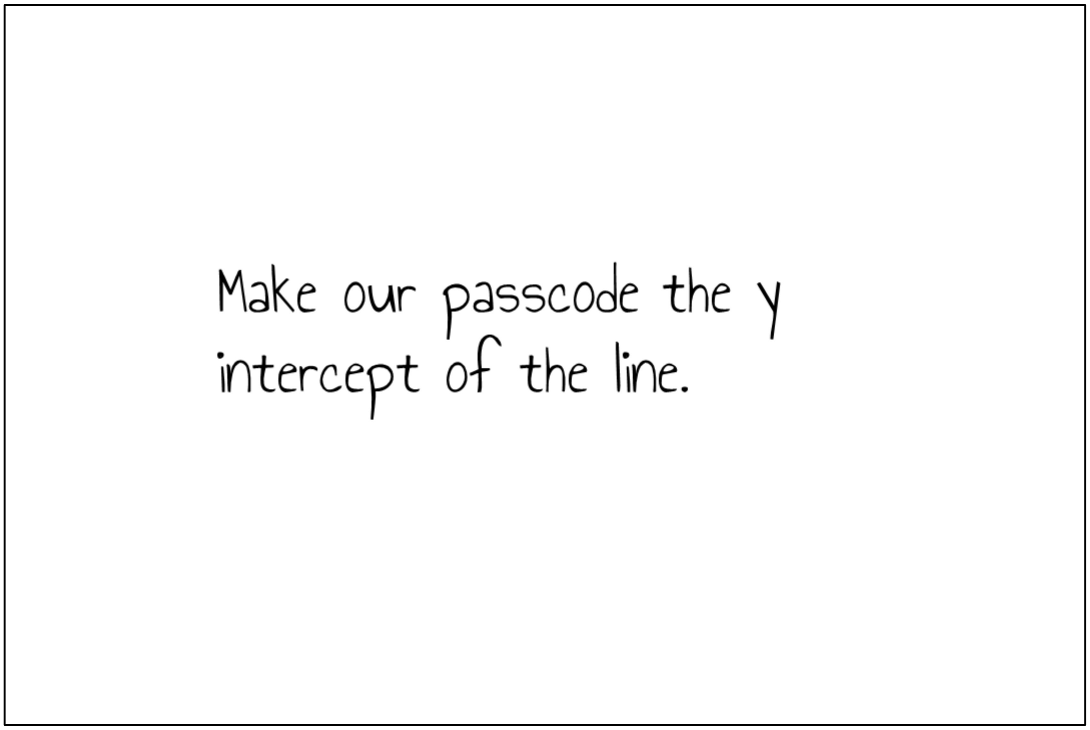


Then you would know exactly which line I was talking about.


QUESTION: How can you design your vault (with secret passcode 1234) so that if ANY two of them put their information together, they'll come up with the passcode 1234 (but any one of them will not know the passcode?)

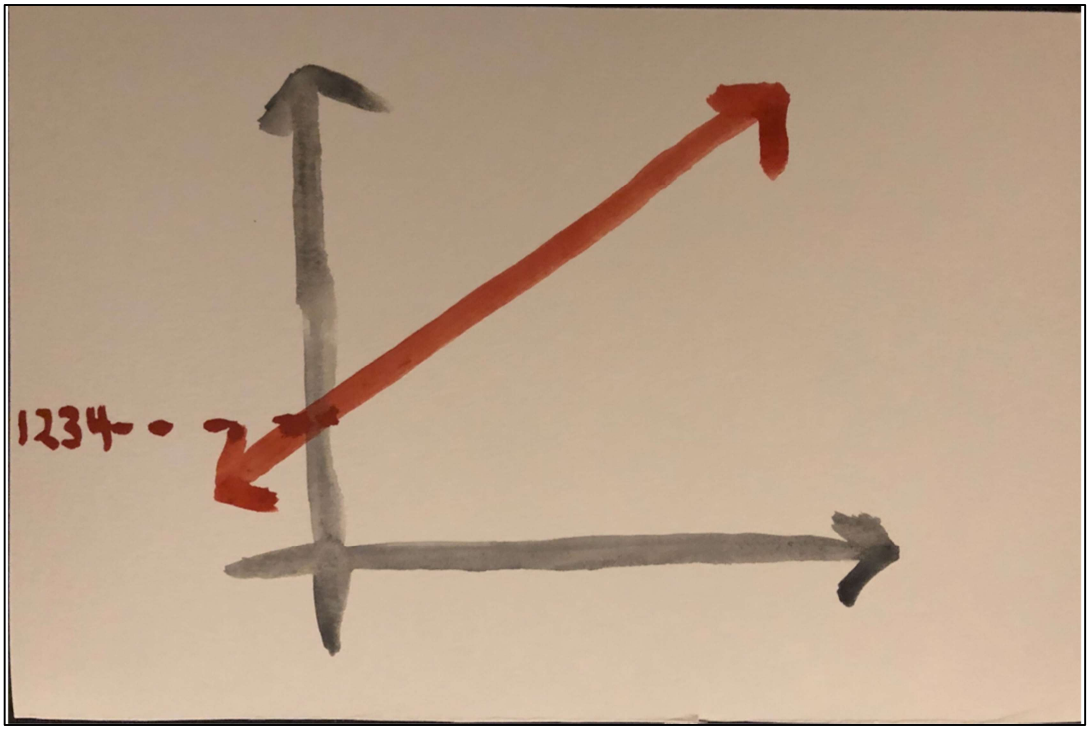
Now let’s go back to our earlier question, but make it a bit more specific — let’s say the passcode is a particular number, say 1234. Any new ideas?

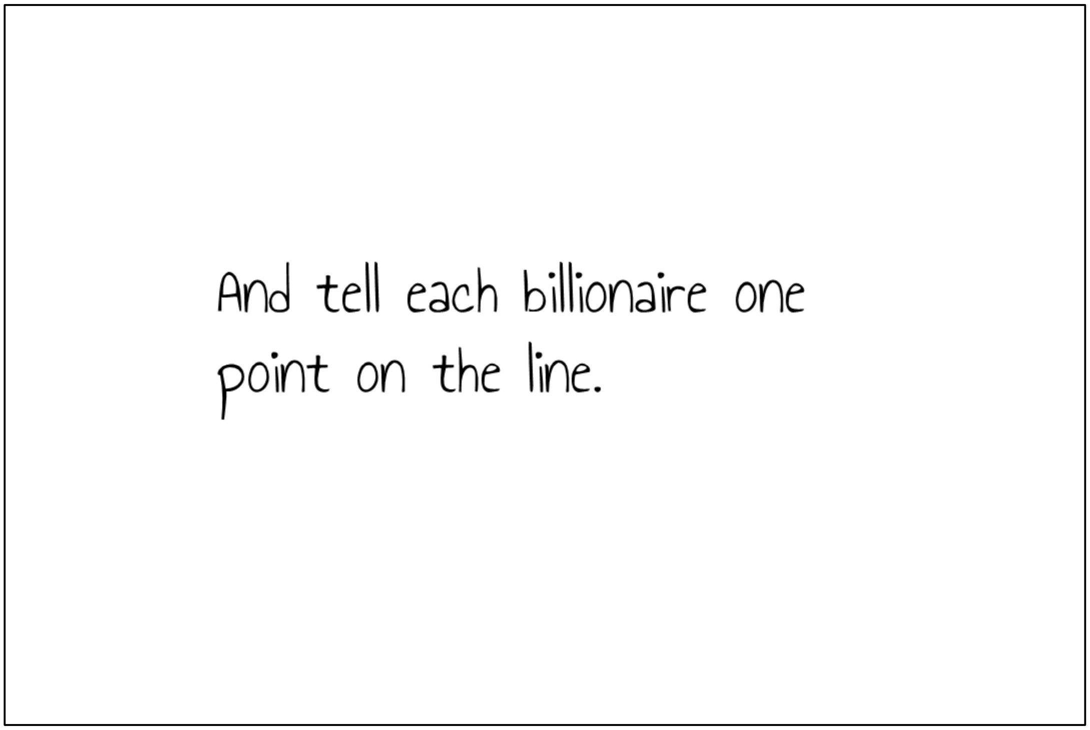
The answer is...



Make our passcode the y intercept of the line.

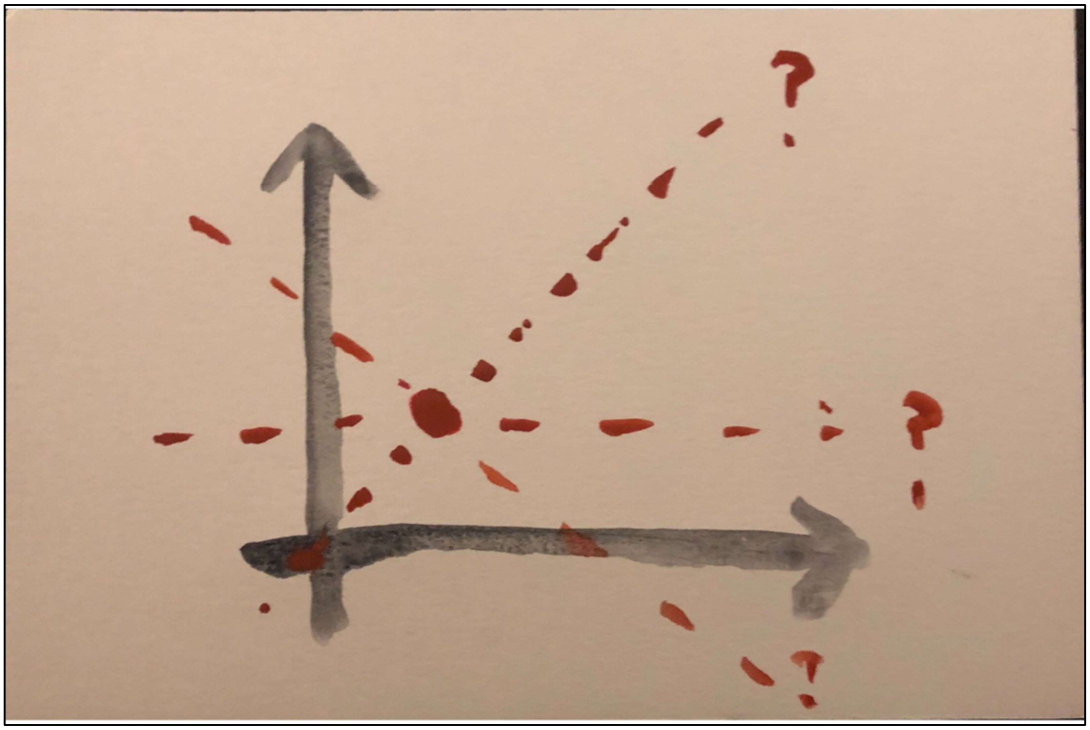
...You can make your secret code (say, 1234) the y-intercept of a line.


1234

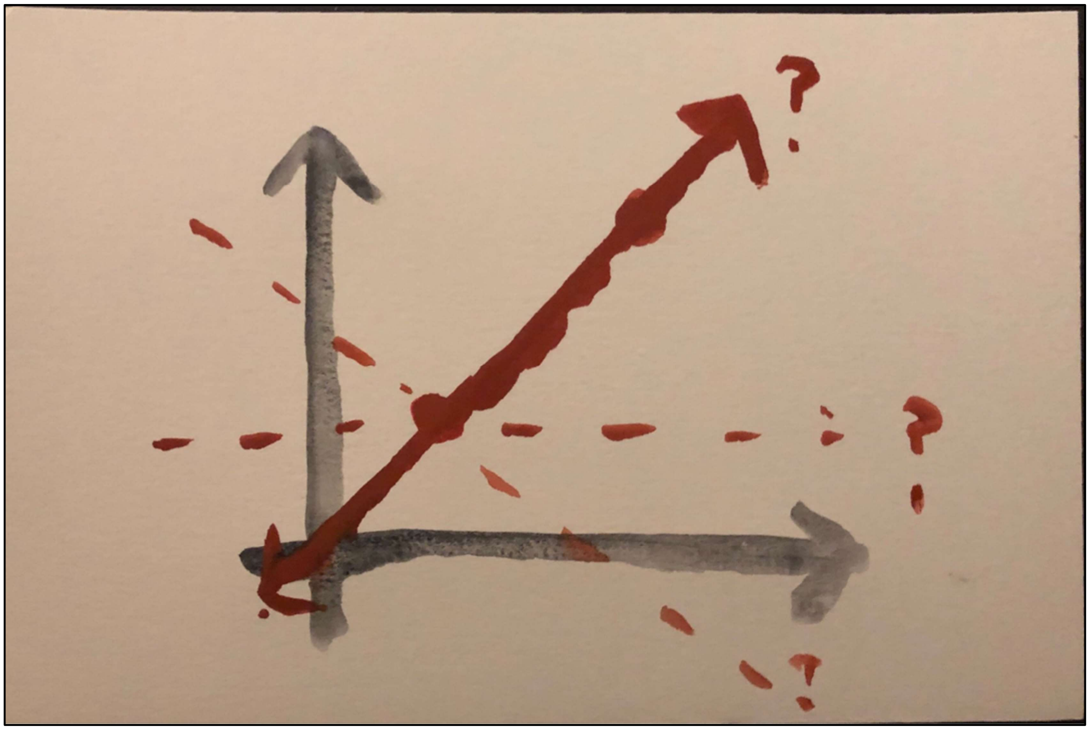


And tell each billionaire one point on the line.

Then, you can tell each of your billionaire friends one point on the line.



Now, one of them alone won’t know which line you’re talking about.

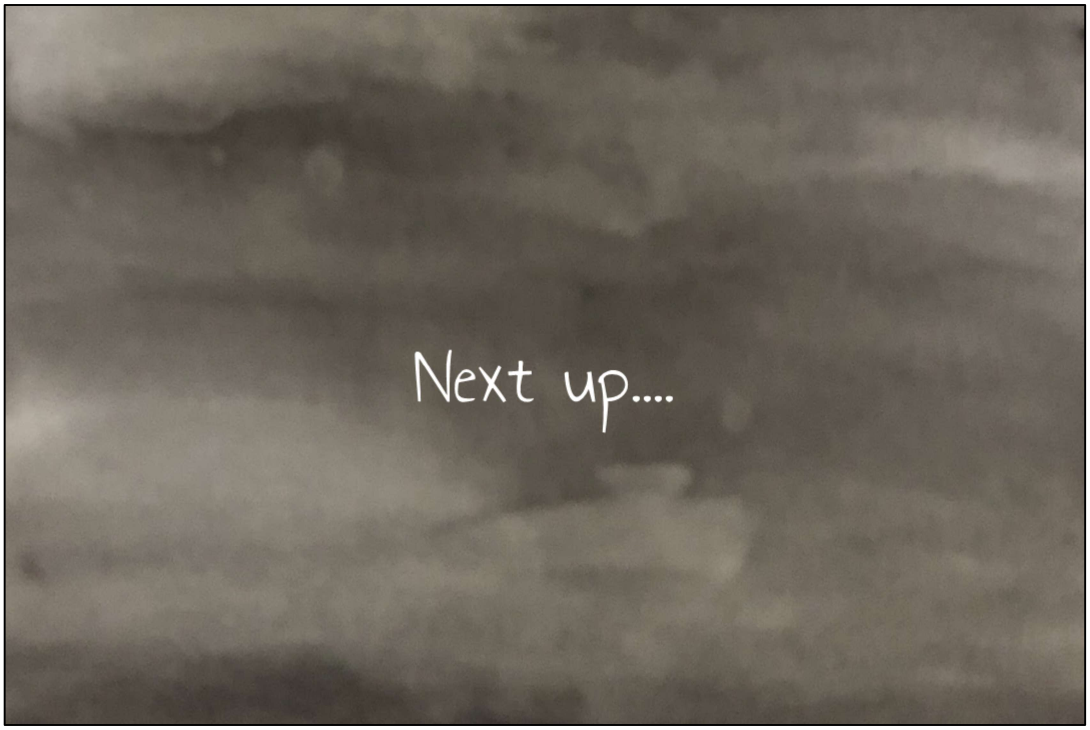


But if two of them put their information together, they'll figure out which line you're talking about, find the y-intercept, and recover your generous inheritance.

Now, I should note that I'm glossing over some details, because computers can only represent decimals with finite precision, and so the number of lines to be represented by a computer isn't truly infinite. And because of that, even just one point may allow a cryptographically savvy billionaire to get closer to finding the y-intercept (the key code). It turns out that using a mathematical object called a finite field, you can make it so that knowing one point gives you absolutely no knowledge about the y-intercept.

It turns out you can apply an analogous technique to any number of billionaires, and any number of people you want to be able to be together to figure out the secret.

I'm going to move on to another problem now, but if you're interested in learning more about these sorts of problems, they're called "secret sharing" problems. The more general version of this algorithm is called *Shamir's Secret Sharing*.

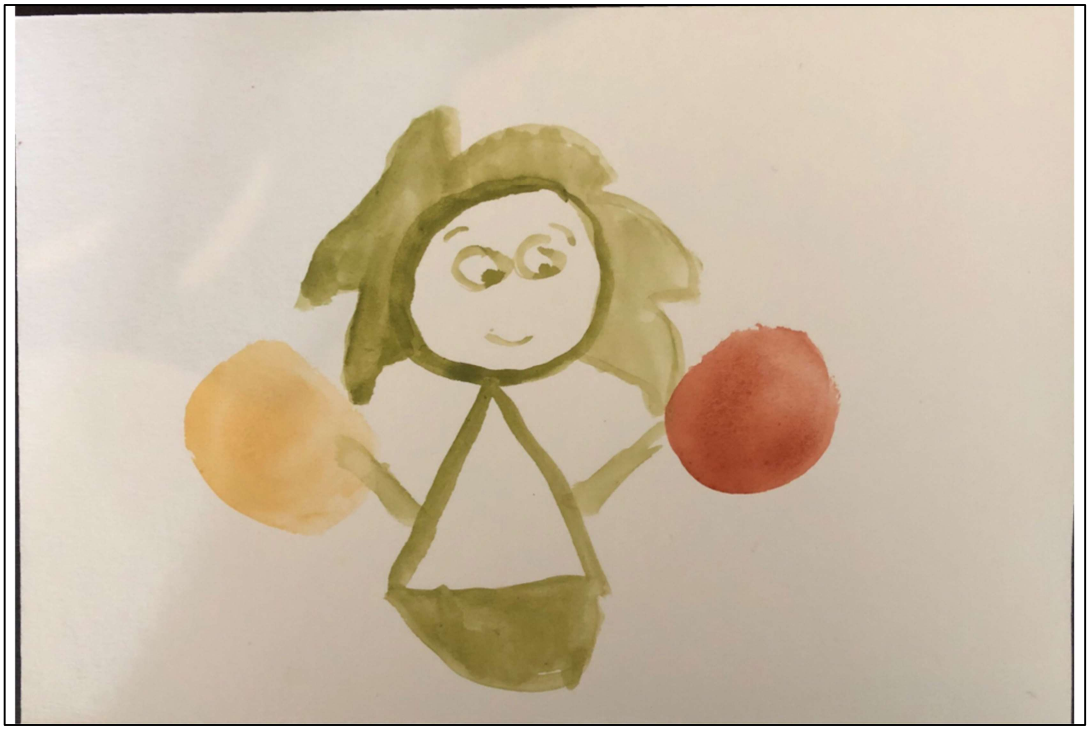


Next up....

I hate transitions, but I do love zero-knowledge proofs.

So let’s say you’re colorblind.

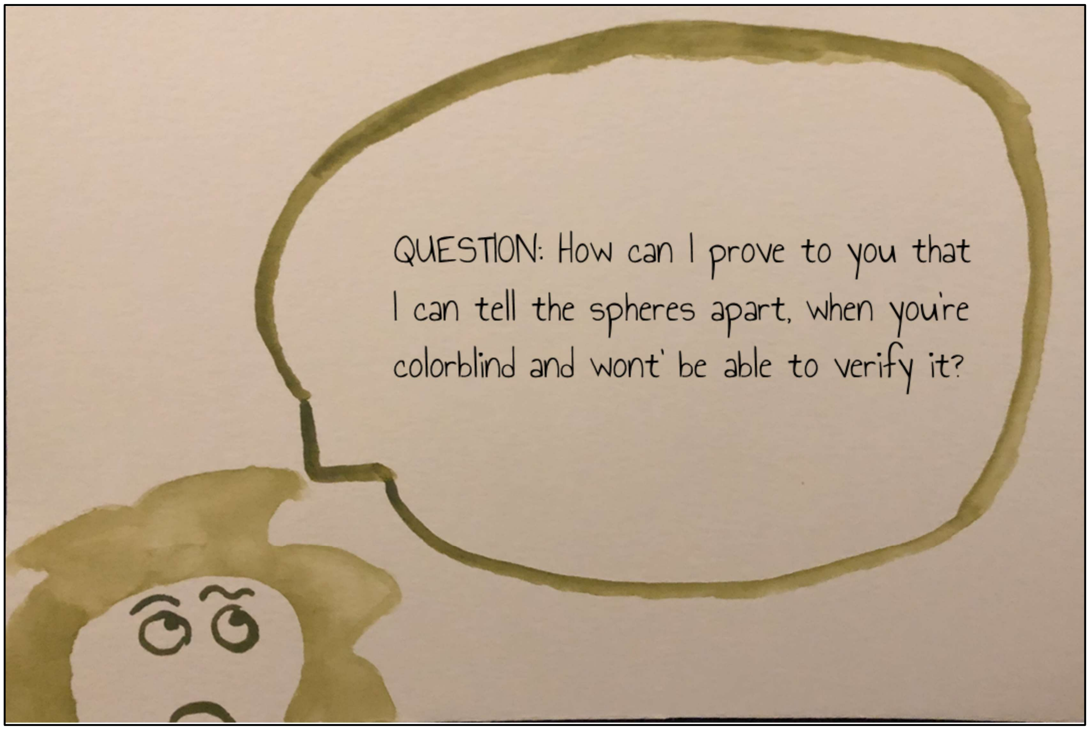
And I — that’s me in green — have two spheres.



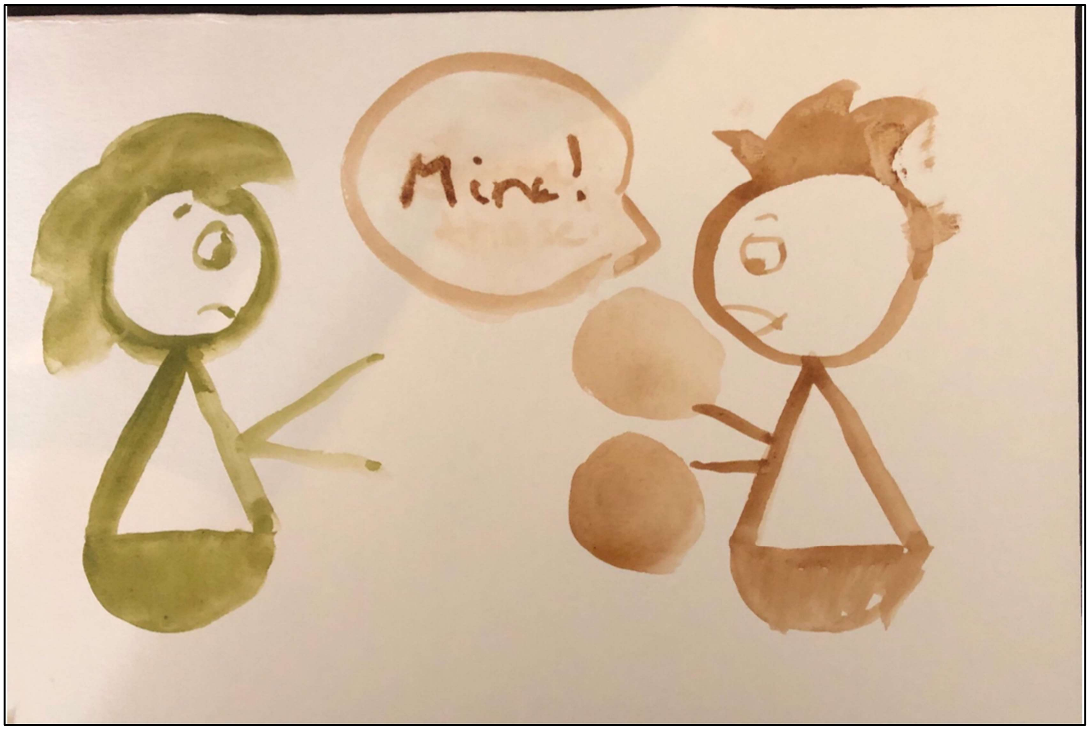
And I insist that these spheres are different colors.

But you — remember that you’re colorblid — can’t tell them apart.

How can I prove to you that i can tell the spheres apart, when you’ll never be able to tell the spheres apart?


QUESTION: How can I prove to you that I can tell the spheres apart, when you're colorblind and wont' be able to verify it?

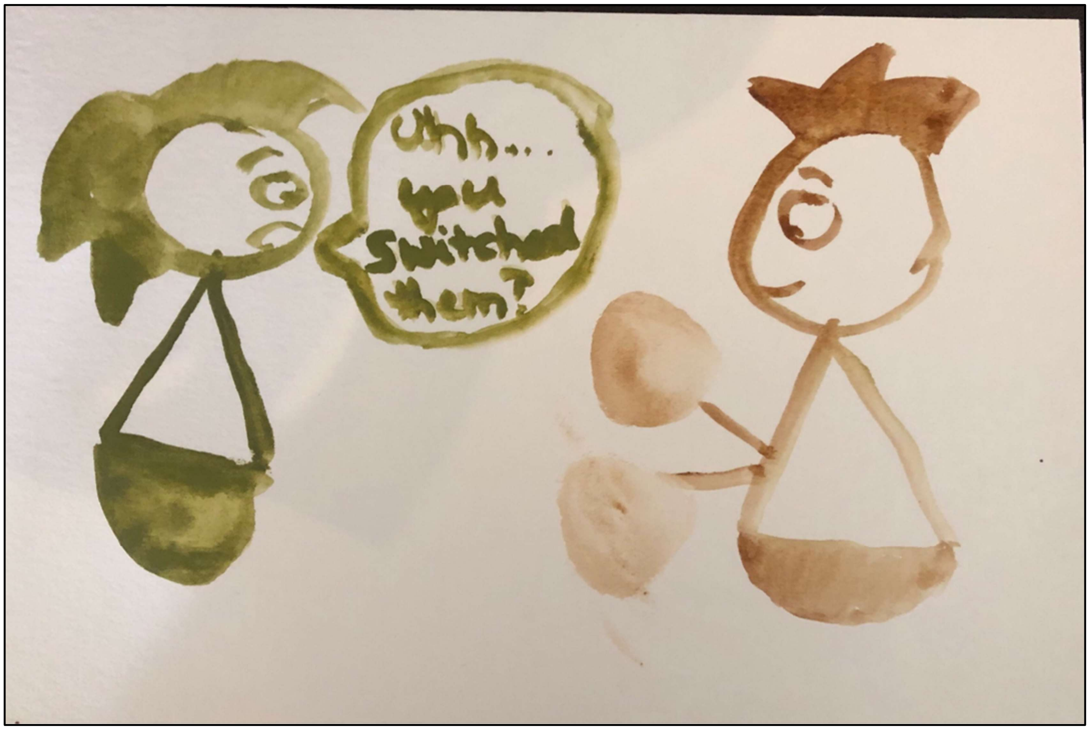
Think about it. If I have two spheres in front of me, that look the exact same, what tests would you put me through to check if I can actually differentiate them?



Well, you can take both the spheres from me...

SWITCH
DON'T SWITCH

And put them behind your back. Then, you can choose to either (1) switch which one is in your left and right hand, or (2) not switch them.

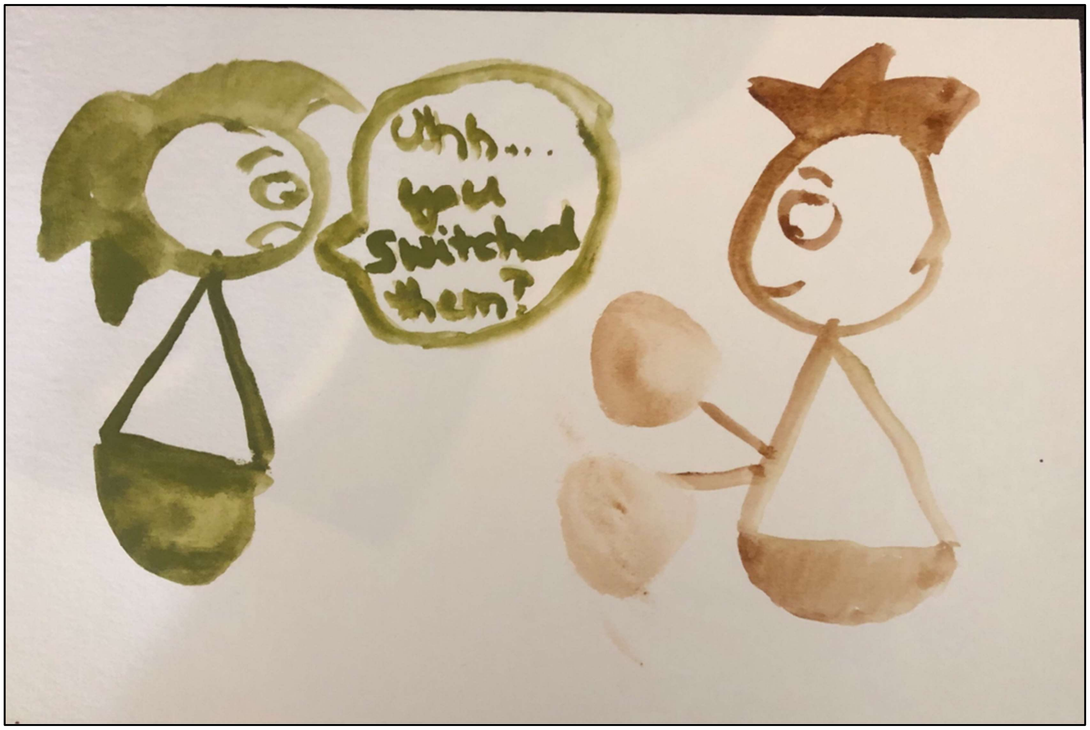


If I can correctly tell whether you switched them or didn't switch them, you know I was telling the truth.

Of course, I could have just guessed and happened to get the answer right (with 50% probability). So you don't know for sure that I can actually tell apart the spheres. At this point, as far as you can tell, there's really only a 50% chance I can actually tell apart the spheres.

SWITCH
DON'T SWITCH

But if you run this test a lot of times...



...and I get the correct answer every time, then you should be more and more certain that I can actually tell apart the spheres.

So this serves as a “proof” that I can tell the spheres apart, even though you yourself can’t tell them apart.

This is the founding question behind zero-knowledge proofs: how can I prove I know something, without ever being able to show you the content of what I know?

In this case, how can I prove I know the colors of the balls, without ever being able to show you the colors of the balls (in a way that you could trust me)? The way I did it was a zero knowledge proof.

In cryptography, zero-knowledge proofs are more commonly thought of as a protocol that allows someone to prove that they *have* a certain piece of information, without actually *revealing* that piece of information. “I know it...but I can’t tell you what it is...but I really do know it.”

So this has been...problems you can solve with cryptography! I hope you enjoyed playing. And thanks for your time.