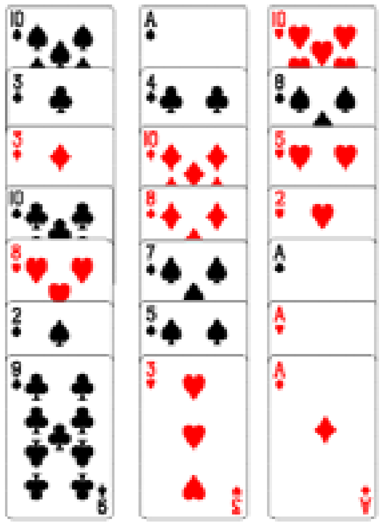
The 21-card trick: the one where you read minds

The magical effect

A volunteer shuffles a pack of cards. You deal out single cards, left to right into three piles of seven cards, all face up and visible. Your volunteer mentally selects one of the cards. You read their mind and tell them the card they are thinking of...

Mind reading of course is not that easy (unless your volunteer is a very clear thinker with a thin skull), so you may need a bit of help.

They mustn't tell you which card it is, but get them to tell you the pile it is in. You collect up the cards, and deal them out a card at a time left to right into three piles once more. Again they tell you the pile their card is in, you collect the cards once more, saying you're struggling to "read their mind". Deal the cards out across the table in the three piles again in the same way. Your friend indicates the pile their card is in. Collect the cards again and deal them into the three piles one last time. You immediately announce their card and magically it is in the very middle position of the pack.



Laying out the 21-card trick

The mechanics

Let's look at the 'mechanics' of the trick: how do you make it work? It involves several deals, each apparently shuffling the order of the cards, but doing so in a rather cunning way. In fact it's really rather simple.

All you have to do is make sure you always put the pile your volunteer selects carefully between the other two piles and deal the pack as above. Do that and after the fourth deal the middle card of the middle pile is the chosen card, which you can reveal as you see fit.
If you are having trouble getting it to work, see our more detailed instructions with pictures at www.cs4fn.org/mathemagic/magicshuffles/
There is even a computer program there that can do the trick itself (and so read your mind over the Internet)!

The 21-card trick: the one where you read minds

The showmanship

Showmanship is important for a good trick. You need some patter to make things more fun and also distract attention from what is really happening. You can come up with your own ideas but here is a version we do.

After first dealing out the cards, stare into the person's eyes as you try and read their mind. Tell them they shouldn't giggle as giggles bubbling up get in the way of the thoughts. (They probably will then struggle not to giggle). Say you need to try again as there were too many giggles. On the second deal try it from the back of the head. After all (you explain) the front of the skull is the thickest part as it is important to protect your brain. Remind them not to giggle… complain it's not working as all they are thinking about is not giggling instead of the card! You will need to deal again. Try this time through their ears – stare hard and you will probably get the colour at least. One more deal and you will have it. Double check through the other ear to make sure it looks the same and you have it! Gradually turn over the ones they weren't thinking of, a few at a time (maybe make a mistake turning over the middle column then correct yourself). Finally their card is the one left face up.

Films We Loved

The Prestige is a great Oscar™-nominated film about the rivalry of Professional Magicians, Science and perhaps(?) supernatural powers.

Learn more at **www.cs4fn.org/mathemagic/**

Magic and computers – developing your own algorithms

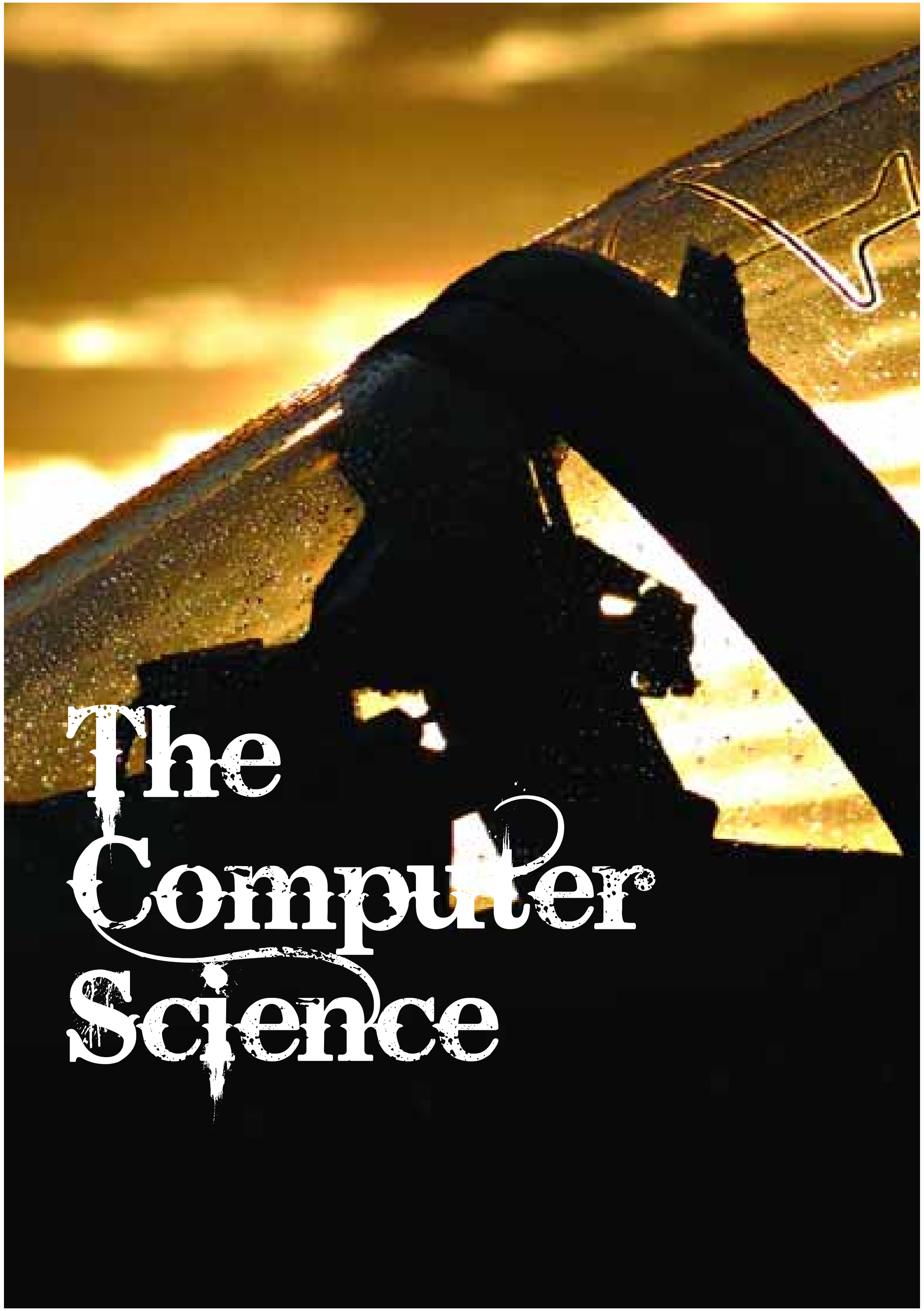
Once you understand the mechanics of a trick and why it works you can play with some ideas. The order of the chosen pile must not be changed, but the two other piles could for example be shuffled before being put together. As long as the chosen pile goes undisturbed between the two other piles of seven cards the order of the other cards doesn't matter. You might want to try and come up with your own additional twists and ways to build them into your presentation now you know how it's done.

Learn more at **www.cs4fn.org/mathemagic/**

A Bit about Magicians

Persi Diaconis was a professional magician, but his passion to debunk crooked casino games pulled him into advanced mathematics. He is now a Stanford professor of Mathematics and Statistics studying the randomness in events such as coin flipping and shuffling playing cards.

He and fellow mathematician David Bayer have shown that you need to give a pack of cards seven dovetail shuffles before the cards are really in a random order.



The 21 card-trick: The Computer Science

Step by step

You want to be sure a magic trick always works. After all, it may work 99 per cent of the time but could you be sure that the one time you're trying to impress a friend or in front of a big audience it would not be the one per cent it didn't work? I know what my luck is like!

Some tricks need your skill at sleight of hand to work. The ones we prefer always work. Computer Scientist's call them **'algorithmic'**. An algorithm is just a clear set of actions to be taken in a given order that achieve some task. Guaranteed!

The steps that you go through to get the 21-card trick to work are like this. They are also similar to the way that a computer steps through its instructions in a software program. All that computers do, in fact, is follow instructions. They follow algorithms that programmers work out for them. The idea is that if they follow the algorithm then they will always complete their task, whether it is playing chess, sending your emails or flying a plane. Every program you have ever used is working the same way as an oversized magic trick.

The point about an algorithm is that if you follow its instructions exactly, you are guaranteed to achieve what you are trying to do…if the algorithm is correct. What if it isn't? Are we really sure our trick always works, whatever?

Testing times

How could we make sure our algorithm is correct and our trick does work? Well we could do the trick lots of times and check it works every time. Computer Scientists call that **'testing'**. It's the main way programmers make sure their programs are correct. They run the program lots of times with different data. Would that be enough to be sure, though?

How many times would we need to do the magic trick to be safe? To be really certain it looks like we would have to try it out with every possible set of 21 cards, in all possible starting positions, checking for every card the person might have thought of.

Try it... How many orders did you do before you got bored? It's a lot of combinations... there are far too many to test them all. It would take an impossibly long time. Similarly testing programs exhaustively like this is not practical. Most programs are far more complicated than this simple trick after all. Instead, as many combinations as possible are tested given the time available. If it works each time then the programmer assumes it works in the cases they didn't try too (and hope!)

That is why there are so often bugs in programs – too much hope, not enough testing!

The 21 card-trick: The Computer Science

There must be a better way!

Perhaps we can be a bit cleverer than that though and work out a shorter set of tests that still give us the guarantee that our trick always works. With a bit of thought it's obvious it doesn't actually matter what any of the cards are. All that matters are the 21 start positions. If a card in the first position ends up in the centre when we test it, we can reason that every time, if a person thought of the card in that position at the start, it will end up in the middle. With this little bit of reasoning we have reduced our testing problem to only 21 tests: one for each starting position. Programmers use similar kinds of reasoning, based on their knowledge of the structure of the program to reduce how many tests they do too.

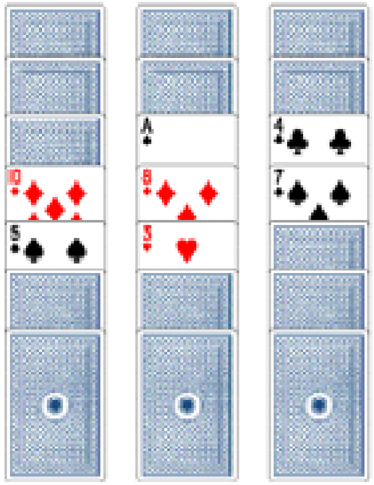
Prove it!

In fact we can go further and do some more reasoning to prove the trick always works. If the proof has no flaws then it proves the trick (or program) works whatever the combination …and you don't need to test any of them. It might be a good idea to still do some testing though. After all, you could have made a mistake in your proof!

It boils down to the fact that putting the chosen pile (column) in the middle of the other two piles and re-dealing the cards in effect limits where the chosen card can go. Let's work through it.

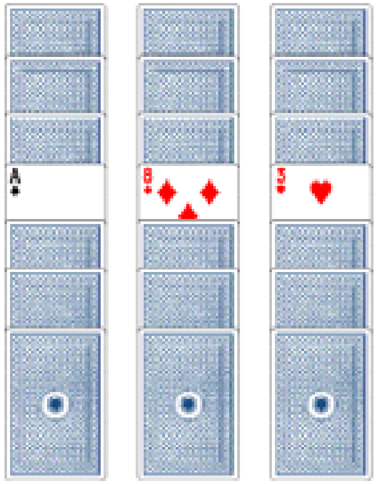
After Deal Number 1: After the first deal of the cards into three piles, the seven-card pile holding the chosen card is put in the middle of the other two. There are now only seven places it could be.

After Deal Number 2



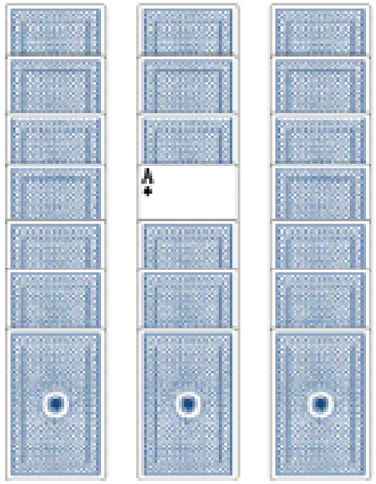
You deal the cards into three new piles. Where do those seven cards from the middle pile go? Anywhere? No. The seven possible places are: the fourth or fifth card of the first pile; the third, fourth or fifth card of the middle pile, or the third or fourth card of the last pile. They are just the middle cards of each pile (as above). The volunteer tells you which pile again, and you again put that pile between the other two. The chosen card must be in the third, fourth or fifth position of the middle pile now. Only 3 possible places are left.

After Deal Number 3



You deal again. This time, the card has to be the fourth card – the middle card – of the first, middle or last pile. Why? There were only three possible places and they each get moved to the middle of their pile as they are dealt out again. In fact more than 40 per cent of the time, it will be in the middle pile (can you see why?), so that's a good pile for you to guess if you want. Once your friend tells you which of the three piles has their card, you know exactly where their card is.

After Deal Number 4



The fourth deal moves the chosen card to the middle of the middle pile… just for effect.

The correctness of algorithms

What we have just done is give a convincing (we hope) argument that the trick or algorithm always works. That is all that mathematical proofs are: convincing arguments where there is no room for doubt if you follow the detail. Here we were just proving that a trick works, but as we saw the instructions of the trick are an algorithm – just like a computer program. It's very important that programs always work too. We can therefore similarly do proofs about the algorithms behind programs. Proofs are just one of the ways computer scientists have developed to help find bugs in programs, and it's useful for finding them in computer hardware too.

Learn more at **www.cs4fn.org/mathemagic/**

A Perfect Shuffle: the one where you magically shuffle a card to a position of your choice

The magical effect

The magicians' art of shuffling in special ways to make tricks, like the 21-card trick, work can also help us build computers. Magicians want to move cards around efficiently; computers want to move data around in their memory efficiently.

In a perfect shuffle, the magician cuts the cards exactly in half and perfectly interlaces them, alternating one card from each half. It takes years of practice to do but looks massively impressive. There are two kinds of perfect shuffles. With an 'out-shuffle' the top card of the deck stays on top. With an 'in-shuffle' the top card moves to the second position of the deck. Magicians know that eight perfect out-shuffles restore the deck to its original order! It looks like the deck has been really mixed up, but it hasn't.

A Perfect Shuffle: The Computer Science

Brent Morris: Magician and Computer Scientist

Computer scientist Brent Morris was fascinated by magic. In particular he became interested in the 'perfect shuffle' in high school and has pursued its mathematics for more than 30 years with some amazing results. He earned his Doctorate in Maths from Duke University, and a Masters in Computer Science from Johns Hopkins University in the United States. He is believed to have the only doctorate in the world in card shuffling. He also holds two US patents on computers designed with shuffles, and has written a book on the subject called *Magic Tricks, Card Shuffling, and Dynamic Computer Memories…* but why so much interest in perfect shuffles?

Binary shifts – as if by magic

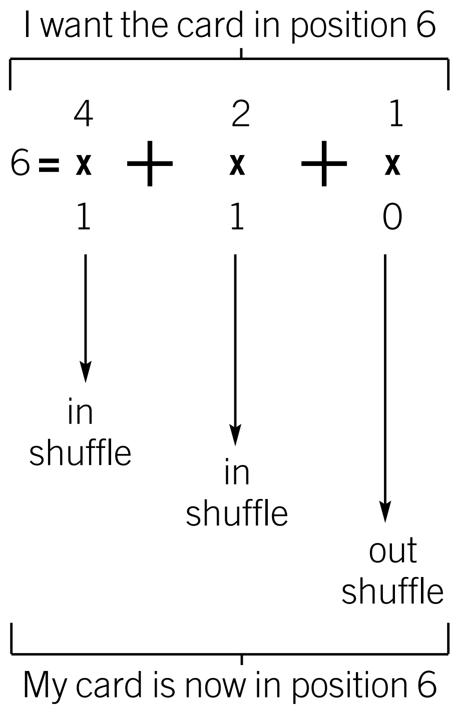
You can use perfect shuffles to move the top card to any position in the pack, using a little bit of the maths behind computers: binary numbers. Suppose you want the top card (let's call that position 0) to go to position 6. Write 6 in base 2 (binary), giving 110 (1x4+1x2+0x1). Now read the 0s and 1s from left to right: 1:1:0. Then, working through the 1s and 0s, you perform an out-shuffle for a 0 and an in-shuffle for a 1. In our case that means:

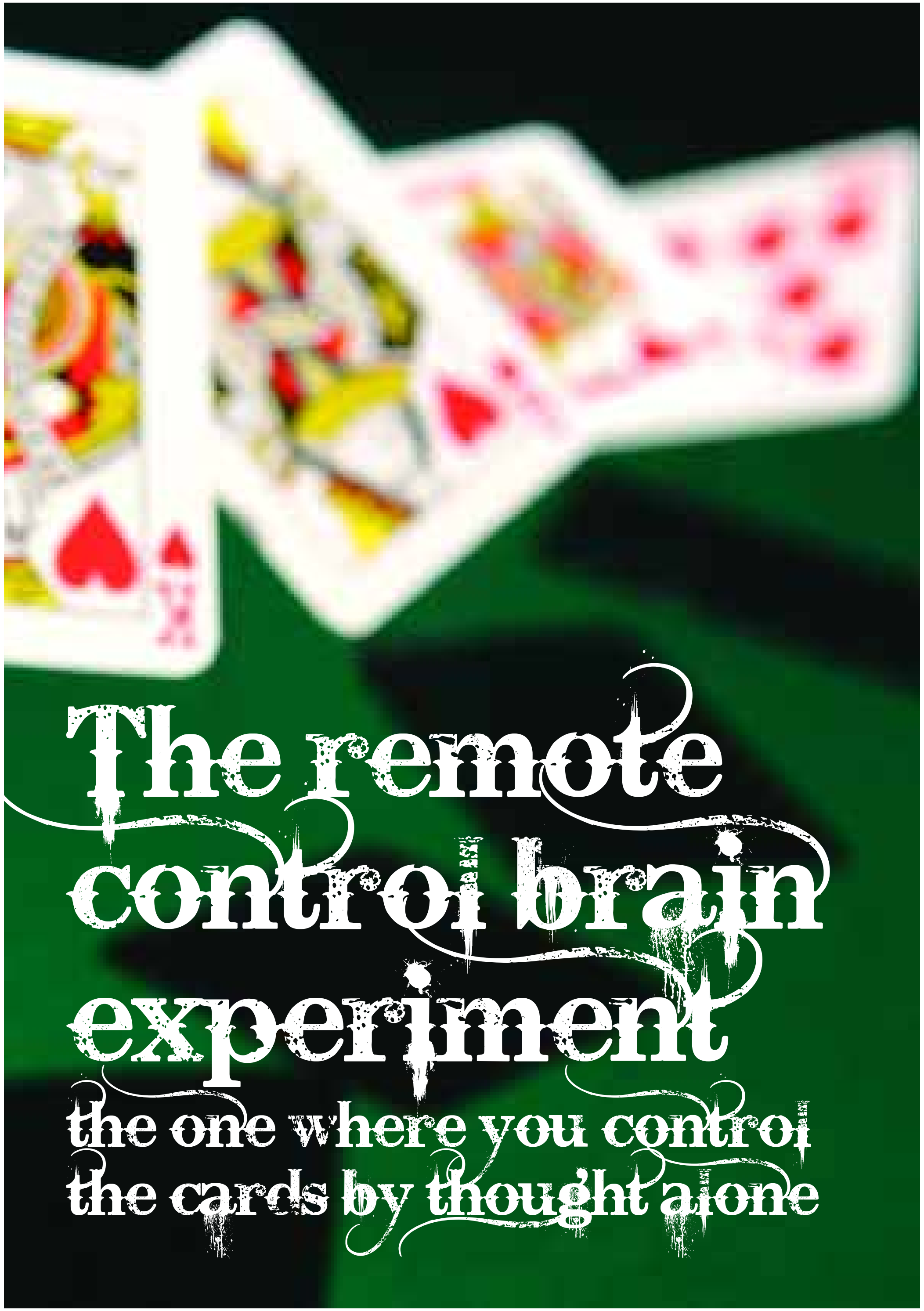
1: an in-shuffle, first

1: another in-shuffle,

0: and finally, an out-shuffle

As if by magic (if you are capable of doing perfect shuffles) the top card will have moved to position 6. Of course it works whatever the number, not just 6. What does this have to do with the design of computers? You can use exactly the same ideas to move data efficiently around computer memory, which is what Brent Morris discovered and patented.





The remote control brain experiment: the one where you control the cards by thought alone

The magical effect

Get a deck of cards and give them a good shuffle. Spread the cards on the table face down. Now think of the colour RED and select any eight cards, then think of the colour BLACK and select another seven cards at random. Now think of RED again, select another six random cards, then finally BLACK again and select five cards.

Shuffle the cards you chose, and then turn the pile face-up. Take the remaining cards, shuffle them and spread them face down.

Now the remote control starts. Concentrate. You are going to separate the cards you selected (and that are now in your face-up pile) into two piles: a RED pile and a BLACK pile, in the following way.

Go through your face-up cards one at a time. If the next card is RED put it in the RED pile. For each RED card you put in your RED pile think RED and select a random card from the face down cards on the table without looking at it. Put this random card in a pile face down in front of your RED pile.

Similarly if the next card is a BLACK card put it face up on your BLACK pile, think BLACK and select a random face down card. Put this face down card in a pile in front of your BLACK pile. Go through this procedure until you run out of face-up cards.

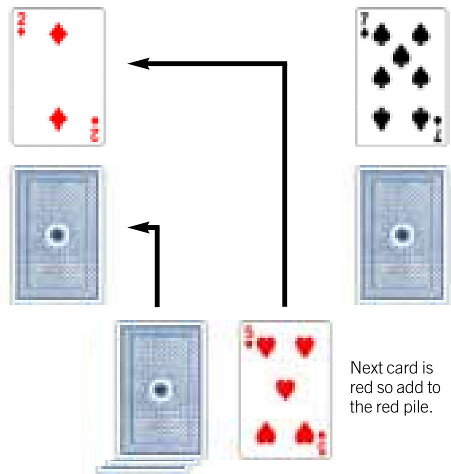
The experiment so far

You now have the following: a RED pile and in front of that a pile containing the same number of face down cards you selected while thinking RED. You also have a BLACK pile in front of which is a pile of random cards you selected while thinking BLACK.

Interestingly your thoughts have influenced your choice of random cards! Don't believe me? Look at the pile of random cards you chose and put in front of your RED pile. Count the number of RED cards in this pile. Now look at the random cards in front of your BLACK pile, and count the number of BLACK cards you selected. You selected the same number of RED and BLACK cards totally at random!

One card out and it wouldn't have worked! It's a final proof that your sub-conscious mind can make you choose random cards to balance those numbers! ... Or is it?

Is mind control a reality? Do you now believe in hocus-pocus? Or are you instead looking for an explanation of why it always works?

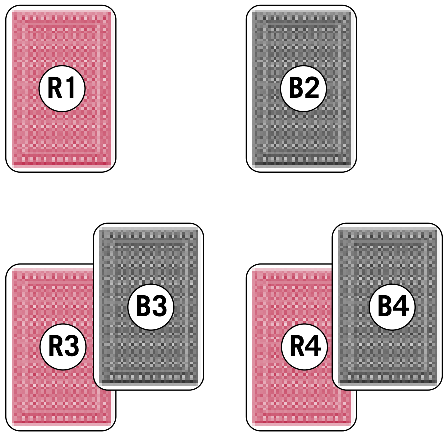


The remote control brain experiment: The Computer Science

Of course it's not mind control. It's mathematics, but you knew that didn't you? I thought you would. But how does this mind reading miracle work? Well it's all down to Abracadabra algebra. Algebra is an area of Maths that matters a lot to Computer Scientists.

The set up – let's get abstract and do some algebra

Pile 1 (RED) Pile 2 (BLACK)



Pile 3 Pile 4

Pile 1 has R1 red cards and nothing else.
Pile 2 has B2 black cards and nothing else.
Pile 3 has R3 red cards and B3 black cards.
Pile 4 has R4 red cards and B4 black cards

Let's call the number of cards in the two piles you dealt R1 for the red pile (pile 1) and B2 for the black pile (pile 2) – see the diagram. The two other piles in front of these contain a random mixture of red and black, so let's say that the pile in front of R1 (pile 3) contains R3 reds and B3 blacks, and the pile in front of B2 (pile 4) contains R4 reds and B4 blacks.

So what do we know?

The first task is to work out what we actually know and turn it into the mathematical equations of the trick.

We actually asked you, in the first part of the experiment, to divide the pack in half. You may have missed that but 8+7+6+5=26.

Now we also know that, for a full pack of 52 cards half (26) are red, and the other half are black so all the red cards add up to 26 and similarly the blacks. We can write that as an equation using the names R1, R3 and R4 for the different sets of red cards and similarly for the black cards. We have to use names because we don't know the actual numbers.

R1 + R3 + R4 = 26
Call this equation (1)

B2 + B3 + B4 = 26
Call this equation (2)

The remote control brain experiment: The Computer Science

We also know the number of cards in the RED pile 1 (R1) is the same as the number of face down cards placed in front of it in pile 3 (made up of R3 red cards and B3 black cards) so together R3+B3 must add up to R1. Similar reasoning holds for the cards in front of the BLACK pile (pile 2 with pile 4). So we know two more equations:

R1 = R3 + B3
Call this equation (3)

B2 = R4 + B4
Call this equation (4)

Now we can start combining these equations by swapping things for their equals. For starters, we know R1 is exactly the same as R3+B3 from equation (3) so if we replace R1 in equation (1) by R3+B3 we get the same thing:

(R3 + B3) + R3 + R4 = 26
Call this equation (5)

Similarly if we substitute equation (4) in equation (2) eliminating B2 we get

(R4 + B4) + B3 + B4 = 26
Call this equation (6)

Combining equations (5) and (6) as both add up to 26, we get

(R3 + B3) + R3 + R4 = 26 = (R4 + B4) + B3 + B4

We can simplify this by grouping the same things together

2xR3 + B3 + R4 = R4 + 2xB4 + B3

We can also subtract R4 and B3 from each side leaving the sides still equal (we did the same to both). That leaves:

2 x R3 = 2 x B4

Finally, we can divide both sides by 2, giving:

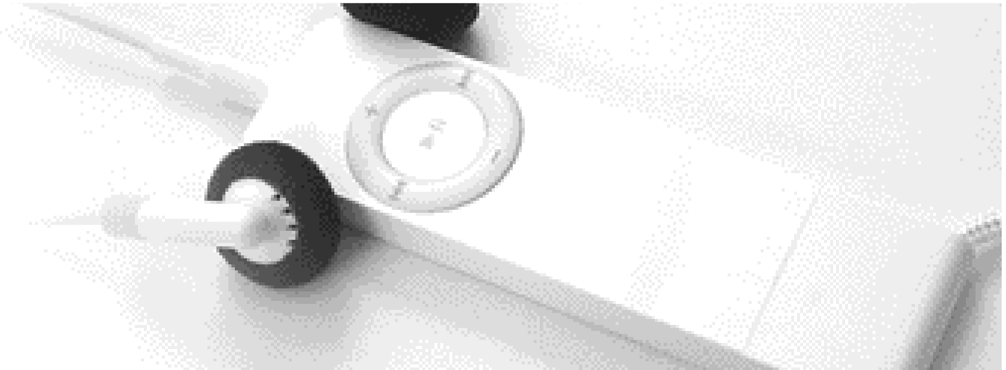
R3 = B4

Back to reality

Now what did we say R3 and B4 stood for? They are just numbers of cards of particular colours in the face down piles.

The maths shows that the number of RED cards (R3) in pile 3 which is in front of the RED pile is ALWAYS equal to the number of BLACK cards (B4) in pile 4 which is in front of the BLACK pile.

That is how the magic works. Maths.



The algebra of self-working magic

The algebra proves the numbers will always be the same. So long as you follow the instructions for the trick (the algorithm) it will always work. The rest of the trick is just presentational flim-flam ... but don't tell anyone how it works!

Algebra is another way that we can prove computer programs will always do what we want them to, by taking the problem and turning it into an **'abstraction'**. As we have done here abstraction uses general quantities such as R1 rather than the actual number of cards, say 12. The use of various kinds of abstraction in programming languages also helps make it easier to write programs in the first place.

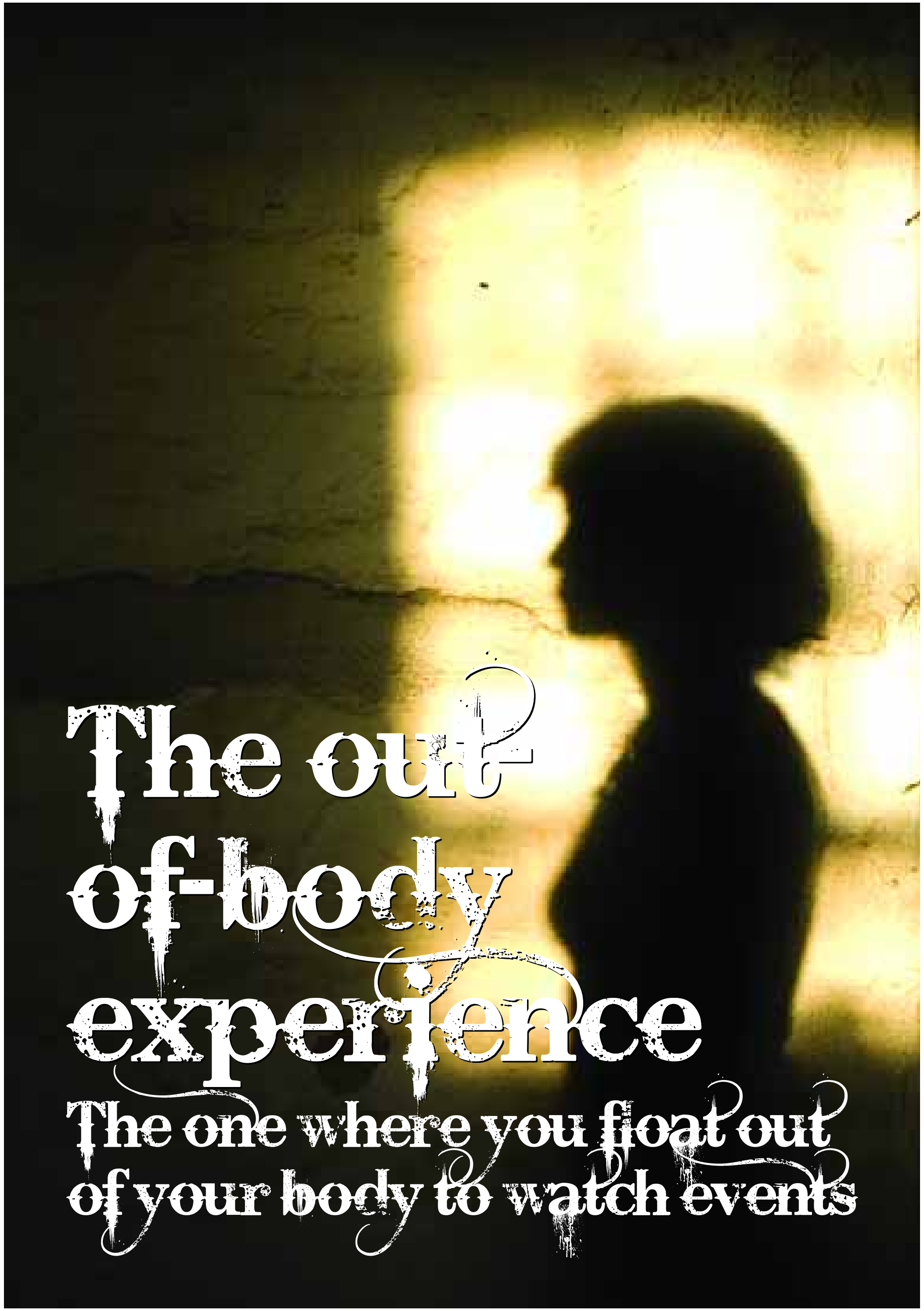
Anyway, using proof, this time algebraic proof, we can be sure that our trick will be self-working without having to try every single set of possible cards, just as we did with the 21-card trick. Remember we need the trick to work 100 per cent of the time if we aren't going to be embarrassed, not 99 per cent of the time.

Now, what if you were talking about, instead of a magic trick, a computer program that was controlling the landing gear on your plane. You would want to be sure that worked 100 per cent of the time as well: that every time the program followed the instructions the right thing happened. Or how about your MP3 player? It is just a computer controlled by programs. It's no good if that only works 99 per cent of the time.

Brain Train: Imagining double digit dexterity

Everyone can do a speedy multiply by 10; you just add a zero to the end of the number. But you can prove your superior mental superpowers by speedy multiplication of a two-digit number by 11. Stretch your imagination and learn how to train your brain's double-digit dexterity by visiting **www.cs4fn.org/mathemagic/** and then challenge your friends.

Would you be happy if every 100th track failed to play? Using similar kinds of abstraction and algebra we can prove programs work correctly too. Mathematical proof is at the core of computer science, and will be increasingly important in the future, helping create safer computer systems, systems you can trust.



The out-of-body experience The one where you float out of your body to watch events

The magical effect

You are blindfolded and lean against the wall at the back of the room with your back to the proceedings. Your spirit leaves your body and flies up to the ceiling so you can watch from above.

Meanwhile, your assistant shuffles a pack of cards. Volunteers then select cards and place them at random either face-up or face-down in a 4 by 4 grid. Your assistant adds more to make it even harder. Your spirit now has a target to watch. A further volunteer then chooses any card from the grid and flips it over. No-one speaks. You are still blindfolded. You can only know which one was flipped if your spirit really is floating above, watching.

You are told to return to your body, which you do. A little dazed, you go straight to the cards and point to the one that was flipped over!