

The Red Black Mind Meld

Age group: 12 – adult (8 – adult if focus on the trick not the proof)

Abilities assumed: None

Time: 30 minutes

Size of group: 1 upwards

Focus

Computational Thinking, Algorithms, Computational Modelling, Abstraction, Logical Reasoning

Syllabus Links

This activity can be used (for example)

- as a general introduction to algorithms reflecting computational thinking from KS2 up, if proof omitted.
- as a general introduction to logical reasoning from KS3 up.
- to show how computational models can be used in real-world problem solving from KS3 up.
- as a practical use of algebra and simultaneous equations from KS3 up

Summary

You do a magic trick where you apparently control a person's actions through the power of thought. A prediction you make about the red and black cards turns out to be true even though no one saw the cards and the person freely picked them at random. You then use abstraction and logical thinking to prove the trick (which is self-working, so an algorithm) always works. You create a mathematical model of the cards and then use algebra to prove a property always holds – the same property that you made the prediction about.

Technical Terms

Algorithms, algebra, mathematical modelling, algorithmic thinking, computational thinking, logical thinking, abstraction, simultaneous equations, safety-critical system, program verification.

Materials

A pack of cards (no Jokers)
Pen and Paper (or slides showing algebra)

What to do

The activity involves doing a simple card trick. You then show how the secret behind the magic can be revealed through computational thinking and algebra. The trick is a fool-proof self-working trick: it works every time with no sleight of hand or misdirection (i.e., it is an algorithm with a magic effect).

The Trick

Take a full pack of 52 cards and give them a good shuffle. Ask for a volunteer to take part in the mind meld. Deal the deck exactly into two equal piles and hand one face down to the volunteer, keeping the other. The volunteer should not look at their cards but you turn your pile face up. Now say you are beginning the mind meld from you to the volunteer. You will control their actions without their knowledge.

Select, at random, a number of **BLACK** cards from your pile and place them in a face up pile in front of you, saying how many **BLACK** cards you have chosen. Ask the volunteer to select the same number of cards (without looking!) from the top, middle, or bottom of their pile, whichever feels right (it will be the mind meld working!) and place the cards in a second pile in front of your **BLACK** pile. Explain this pile will be influenced by your **BLACK** pile.

Now repeat the process but take a random number of **RED** cards from your hand and place them in a face up pile next to your black pile, telling the volunteer how many **RED** cards you have chosen. Again, ask the volunteer to select the same number of cards from anywhere in their pack. They should place them in front of your **RED** pile. Explain that this new pile will be influenced by your **RED** pile.

Repeat the process, alternating between choosing **BLACK** cards and choosing **RED** cards until you have run out of cards. If done correctly the volunteer will run out of cards at the same time. Ask the audience to tell you how many cards to pick for some of your choices.

You now have the following:

- A face up **RED** pile and in front of that a pile containing the same number of face down cards the volunteer selected while under the mind meld.
- A face down **BLACK** pile in front of which is a pile of random cards the volunteer selected again while under the mind meld.

Explain that your mind meld influenced the volunteer's choice of random cards and you can prove it. Even though the pack was shuffled and the volunteer chose cards at random, you used the mind meld to make sure *they put exactly the same number of red cards in front of your red pile as they put black cards in front of your black pile.*

Ask the volunteer to first take the pile of face down random cards in front of your **RED** pile and count the number of **RED** cards in it on to the table. Confirm the number of **RED** cards there. Then ask them to take the random face down cards in front of your **BLACK** pile, and count out loud the number of **BLACK** cards there on to the table.

Through the mind meld you caused the volunteer to select the same number of **RED** cards in the **RED** influenced pile as **BLACK** cards in the **BLACK** influenced pile totally at random! One card out and it wouldn't have worked!

Is mind control a reality? Is it final proof that a suitably powerful mind can control the sub-conscious of another? Did you really make the volunteer choose cards to balance those numbers?

Modelling the Trick

Of course it's not mind melding. It's just computation and mathematics. The trick is self-working. If you follow the steps, it is bound to work. But how can we be sure that is true? We could do the trick lots of times (testing it) and see if it does, but perhaps the next time it won't work. We can't be sure. Instead, with some computational thinking, we can show that what happened is guaranteed to always happen as long as you follow the steps of the trick exactly. We need several aspects of computational thinking. We are of course talking about **evaluation** of the trick – checking that it really does work. To do that we need to use **abstraction** and **logical thinking** and some mathematics that goes with them: algebra. We need to use abstraction to create a mathematical model of the trick and then use logical thinking in the form of algebra to prove that it always works.

Abstraction

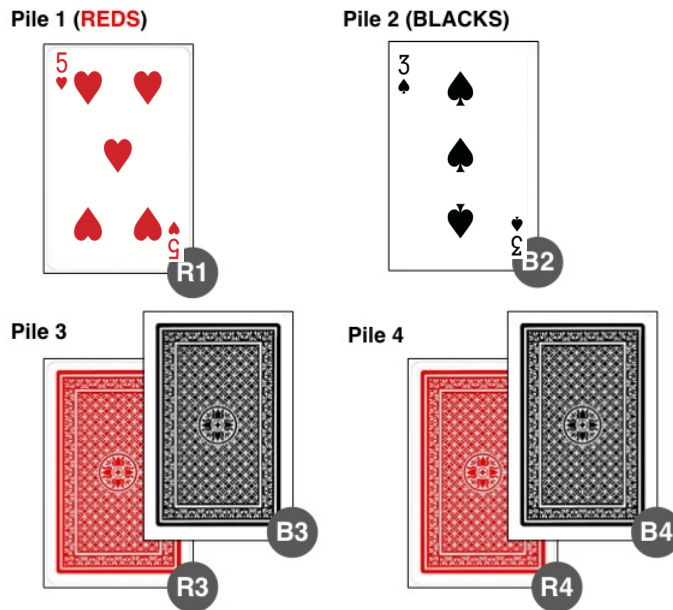
The first step is to create a model of the trick. That involves **abstraction**. Abstraction just means focus only on the information about the trick that matters, while hiding all the detail that doesn't. A model is just an abstraction of a part of the world we are interested in. This trick is about the numbers of red and black cards. The actual values and suits of the cards doesn't matter at all. We can ignore that and just focus on the numbers of each colour that end up in each pile.

Unfortunately, we have no idea how many reds or blacks ended up in any of the piles as the pack was shuffled. We can't actually put numbers to anything (apart from knowing there are no black cards in the face up red pile and no red cards in the face up black pile). Instead we just give a name to the unknown numbers of red and black cards in each pile.

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.

Logical Thinking: The facts

The first task is to use logical reasoning to work out what you actually know and turn it into mathematical equations about the trick.



A full pack of 52 cards was used. It contains **26 RED** cards and **26 BLACK** cards. If you add up the red cards in the four piles it will come to 26. We can write that as an equation using the names R1, R3 and R4 for the different sets of red cards in each pile as in our diagram. We can do the same thing for the black cards. We have to use the names because we don't know any of the actual numbers.

Equation 1:
 $R1 + R3 + R4 = 26$

Equation 2:
 $B2 + B3 + B4 = 26$

Now we know something else about the numbers. Whenever we put some number of cards in the red pile, we put exactly the same number of cards in the face down pile in front of it. That means that the two piles end up with the same number of cards (though again we have no idea what that number is). In terms of our names that means we 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:

Equation 3:
 $R1 = R3 + B3$

Equation 4:
 $B2 = R4 + B4$

Logical Thinking: Some rigorous reasoning

Now these equations don't immediately seem to help, but when a mathematician sees equations that share the same variables (the names we used for numbers) she will start to simplify them by substituting for equals.

So we 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 we can replace **R1** in **Equation 1** by **R3+B3** you get **Equation 5** (see below). Similarly if we substitute **Equation 4** in **Equation 2** eliminating **B2** we get **Equation 6**.

$$\text{Equation 5:} \\ (R3 + B3) + R3 + R4 = 26$$

$$\text{Equation 6:} \\ (R4 + B4) + B3 + B4 = 26$$

Combining **Equations 5** and **Equation 6** as both add up to **26**, we get:

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

That looks horrible! But we can simplify it by grouping the same things together:

$$2 \times R3 + B3 + R4 = R4 + 2 \times B4 + B3$$

Also we have a **R4** and a **B3** on both sides of this equation. We can subtract both **R4** and **B3** from each side leaving the sides still equal (as we did the same to both sides). That leaves:

$$2 \times R3 = 2 \times B4$$

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

$$R3 = B4$$

We have shown that as long as the four facts we started with all hold then the number **R3** will always be the same as the number **B4**.

Logical Thinking: Back to the real world

Now that is all well and good, but what does it mean? We have to return to the real world and see what those variables actually represent. **R3** and **B4** were actually just names that stood for numbers of cards of particular colours in the face down piles. **R3** is the number of red cards in the red influenced pile. **B4** is the number of face down black cards in the black influenced pile. The maths has shown that it is guaranteed that the number of **RED** cards **R3** in **Pile 3** is **ALWAYS** equal to the number of **BLACK** cards **B4** in **Pile 4**. That is how the magic works. Maths.

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 presentation! (Though that has to work too if the trick is to seem 'magical'.

Verifying Programs

Magic tricks are just algorithms but then so are programs. If we can prove that a magic trick always works, then we can prove that a program does too in a similar way: by using the same kind of computational thinking. We take what the program does and use **abstraction** to create a mathematical model of it. We then use logical reasoning and algebra to prove properties of our model. Those properties then tell us facts about the effect of the program in the real world. For example we might prove that if the program controlling a medical device was set to deliver v milligrams of a drug over t hours then that is what it would always do. For a program that helps land a plane we might prove that when the pilot presses a button to lower the landing gear, the wheels are locked into position within in a minute. This is called program verification.

Using algebraic proof, we could be sure that the trick will be self-working without having to try every single set of possible cards. The trick needs to work 100% of the time, not 99% of the time, as that would mean sometimes we would look stupid in front of a live audience. Now what if, instead of a magic trick, we were talking about that computer program controlling the landing gear on your plane, or the amount of a drug being pumped into your arm in hospital. You would want to be sure that they worked 100% per cent of the time as well: that every time the program followed the instructions the right thing happened. It's no good if that only works 99% of the time. Killing someone 1% of the time is not good enough! For safety-critical systems testing the program isn't good enough, you must use logical thinking and rigorous argument too. You need to use mathematics.

Variations and Extensions

Create your own trick

As long as the 4 facts as described in the four equations remain true, you can change the steps and presentation of the trick and it will still work. For example, once all the cards have been put into piles you can swap a card from one face down pile for a card from the other perhaps claiming you lost the mind meld for a moment. Have the class think up their own variations.

Write the algorithm

Have the class write their own crib sheets for doing the trick (i.e., write down the algorithm). They will have to make sure they include all steps in enough detail and get them in the right order. They may also want to use decomposition to split their description in to the main steps and then separately describe each step in more detail.

DIY algebra

Rather than doing the proof of the trick yourself, for a class comfortable with algebra, help them set up the model and initial known facts, then let them try and prove the trick works themselves.

Links to other activities

Get the following activities with links to maths from teachinglondoncomputing.org

The Australian Magician's Dream

Do a magic trick where you predict a card chosen that even the person choosing couldn't have known. Challenge the audience to work out how it is done, teach them how to do the trick and then use it to explain algorithms, searching, and logical reasoning.

Punch card searching

Find out how binary numbers allow you to pull any card you want from a pile. This demonstrates a simple divide and conquer algorithm for searching, and in particular how early computers could quickly pull out any punch card from a pile. See how the algorithm is identical to the Australian Magician's dream magic trick.

Tour Guide

Devise a tour that gets a tourist from their hotel to all the city sights and back to their hotel.

Show how the representation of a problem can alter how easy it is to do and how apparently different problems can be generalized to reveal that they are the same. Show how graphs can be used as a powerful representation.

Live demonstration of this activity

Teaching London Computing give live sessions for teachers demonstrating this and our other activities. See <http://teachinglondoncomputing.org/> for details. Videos of some activities are also available or in preparation.

Acknowledgements

This activity sheet was funded by a grant (ESD23/2014) from the Department for Education, with support from Google and the EPSRC funded CHI+MED project (EP/G059063/1). Teaching London Computing is funded by the Mayor of London.

Use of this material



[Attribution NonCommercial ShareAlike](https://creativecommons.org/licenses/by-nc-sa/4.0/) - "CC BY-NC-SA"

This license lets others remix, tweak, and build upon a work non-commercially, as long as they credit the original author and license their new creations under the identical terms. Others can download and redistribute this work just like the by-nc-nd license, but they can also translate, make remixes, and produce new stories based on the work. All new work based on the original will carry the same license, so any derivatives will also be non-commercial in nature.



Department
for Education

SUPPORTED BY

MAYOR OF LONDON

COMPUTING AT SCHOOL
EDUCATE · ENGAGE · ENCOURAGE

