Magische Informatik

Informatik? – Für mich alles Hexerei!
Informatik? – Finde ich bezaubernd!

von OStR Ulrich Kiesmüller

Äußerungen bezüglich Informatik als Wissenschaft und als Unterrichts- bzw. Studienfach liegen weit gestreut zwischen diesen beiden Aussagen. Die Didaktik der Informatik hat sich der Aufgabe angenommen, Informatik so zu vermitteln, dass sie eher als „bezaubernd" denn als „Hexerei" empfunden wird. Basierend auf dem Projekt „Magic Computer Science" der Queen Mary Universität in London (s. McOwan/Curzon und McOwan/Curzon/Black), deren Partner in diesem Bereich die Friedrich-Alexander-Universität Erlangen-Nürnberg ist, werden in diesem Artikel Zaubertricks aus dem „Mentalisten"-Bereich dazu verwendet, informatische Sachverhalte anschaulich darzustellen. So kann mit einigen in der Informatik häufig eingesetzten Verfahren die „Funktionstüchtigkeit" mancher Tricks nachgewiesen werden, andere wiederum stellen vereinfachte Modelle komplexer Anwendungsbeispiele aus der Informatik dar. Dadurch kann eine bessere Memorierbarkeit der informatischen Inhalte erreicht werden. Jeder einzelne Trick eignet sich als motivierender Einstieg in eine Unterrichtsstunde oder in eine thematische Vorlesungseinheit. Auch zur Veranschaulichung bestimmter Sachverhalte im Übungsbetrieb lassen sich die Beispiele einsetzen. Die Intensität der informatischen Erläuterungen muss je nach Bedarf und Absicht variiert werden. In diesem Artikel wird jeweils zuerst die Vorführung des Tricks und der Effekt für die Zuschauer beschrieben und anschließend der Hintergrund aus Sicht der Informatik ausführlich erläutert.

Bild 1: Karten mit gewählter Herz 10

21 Karten

Es werden 21 gemischte Spielkarten in Form einer 3x7-Matrix wie in Bild 1 dargestellt mit der Bildseite nach oben ausgelegt. Ein Zuschauer wählt – für das Restpublikum sichtbar, für den Mentalisten nicht – eine Karte (in Bild 1 die Herz 10). Der Magier versucht nun durch „Gedankenlesen" herauszufinden, um welche Karte es sich handelt. Da dies nicht zu gelingen scheint, bittet er um eine kleine Hilfe und lässt sich die Spalte angeben, in der die Karte sich befindet (Spalte 3 in Bild 1). Nachdem der Zauberer die Karten spaltenweise zusammengeschoben und wieder zu einem Stapel aufeinander gelegt hat, teilt er sie erneut zeilenweise als 3x7-Matrix aus. Insgesamt lässt er sich dreimal die (nicht notwendiger Weise immer gleiche) Spalte zeigen. Bei der zuletzt mit der Bildseite nach unten ausgelegten Matrix identifiziert der Mentalist dann die mittlere Karte als die vom Zuschauer gewählte und er kann auch deren Wert vorhersagen noch bevor sie zur Überprüfung dem Publikum gezeigt wird. Beim Einsatz in einer Lehr- oder Übungsveranstaltung wird an dieser Stelle der Übergang zur Informatik dadurch motiviert, dass die Lernenden erfahren wollen, wie der Trick „funktioniert“, was bedeutet, warum das beschriebene Verfahren immer (und nicht nur sehr wahrscheinlich) zum gewünschten Ergebnis führt.

Dieser Trick basiert auf dem in der Informatik in verschiedensten Teilbereichen (z. B. rekursive Algorithmen und Datenstrukturen) eingesetzten „Teile und Herrsche“-Prinzip. Im schulischen wie auch universitären Bereich ist eines der bekanntesten Beispiele die binäre Suche in geordneten Listen.

Im Unterschied hierzu wird bei diesem Trick allerdings bei jedem Durchgang jeweils eine Drittelung der Suchmenge durchgeführt. Gibt der Zuschauer in der Ausgangssituation an, dass sich (wie in Bild 1) die gewählte Karte in Spalte 3 befindet, so muss nur noch in den 7 Karten der dritten Spalte weiter gesucht werden. Um diese Verkleinerung der Suchmenge zu kaschieren werden die Karten der anderen beiden Spalten (in denen nicht mehr gesucht werden muss) oberhalb und unterhalb der Karten aus Stapel 2 zum Stapel gelegt. Beim erneuten (zeilenweisen!) Auslegen befindet sich die noch relevante Suchmenge folglich in den mittleren drei Zeilen (vgl. Bild 2). Die jetzt folgende erneute Angabe der Spalte, in der sich die Karte jetzt befindet (Spalte 1 in Bild 2), führt zu einer weiteren Drittelung der zu durchsuchenden Karten, der Umfang der ursprünglichen Suchmenge wurde insgesamt bereits geneuntelt. Auch in diesem Schritt wird die weitere Reduzierung der zu betrachtenden Karten kaschiert, in dem die Karten der nicht gewählten Stapel wie oben vor und hinter den gewählten Stapel gelegt werden, so dass sich die restliche Suchmenge von inzwischen maximal 3 Karten in der Mitte des Gesamtstapels befindet. Dies ist gleichbedeutend damit, dass beim erneuten zeilenweisen Auflegen die gewählte Karte nur noch in der mittleren der sieben Reihen enthalten sein kann. Wird jetzt erneut die Spalte angegeben, in der die gewählte Karte liegt, dann ist sie eindeutig identifiziert. Die ursprüngliche Suchmenge wurde durch diese letzte Drittelung auf ein 27-tel reduziert. Der 27. Teil von 21 Karten ist keinesfalls mehr als eine Karte und somit ist die gesuchte Karte bereits identifiziert. Werden jetzt wie in den beiden ersten Durchgängen die „Restkarten“ wieder vor und hinter die Karten der gewählten Spalte gelegt, wird beim erneuten zeilenweisen Auflegen die bereits identifizierte Karte immer in der Mitte der mittleren Zeile liegen und kann somit auch im verdeckten Zustand identifiziert werden.

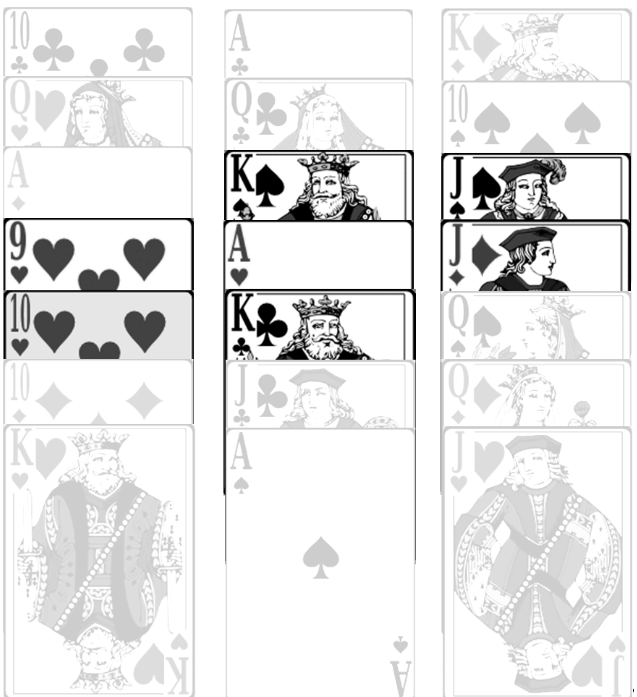

Bild 2: Karten nach zweitem Auflegen

Damit ist hier sowohl die Terminierung als auch die Korrektheit des Algorithmus nachgewiesen, wenn auch nicht mit formalen sondern lediglich mit anschaulichen Methoden. Die Notwendigkeit des Nachweises an dieser Stelle ist für die Lernenden einleuchtend, denn es ist offensichtlich, dass kein Mentalist diesen Trick einsetzen würde, wenn er nicht sicher funktionieren würde. Ebenso Bedarf es keiner Erläuterung, dass der Nachweis wegen des zu hohen zeitlichen Aufwands nicht durch das schrittweise Ausprobieren aller möglichen Ausgangssituationen (hier: gewählten Karten) durchgeführt werden kann. Im Anschluss an diesen Nachweis kann nun entweder der Einstieg in die rekursive Datenstruktur der Liste erfolgen oder der Einstieg bzw. die vertiefende Übung zum Thema Anwendung des „Teile und Herrsche“-Prinzips zur effektiven Suche. Die Idee der Drittelung der Suchmenge lässt sich dann z. B. wie folgt implementieren:

```
public int ternaersuche(int links, int rechts, int
wert){
 grenze1 = links + (rechts - links - 1) / 3;
 grenze2 = links + 2*(rechts - links - 1) / 3;
 if ((links > rechts) || (links == rechts &&
 Feld[grenze1] != wert) || (links == rechts &&
 Feld[grenze2] != wert))
  {return -1;}
 else if (Feld[grenze1] == wert)
  {return grenze1;}
 else if (Feld[grenze2] == wert)
  {return grenze2;}
 else if (Feld[grenze2] < wert)
  {return ternaersuche(grenze2 + 1, rechts, wert);}
 else if (Feld[grenze1] > wert)
  {return ternaersuche(links, grenze1 - 1, wert);}
 else
  {return ternaersuche(grenze1, grenze2, wert);}
```

Auch Betrachtungen hinsichtlich der Komplexität dieses Suchverfahrens lassen sich den Lernenden an Hand des Trickablaufs anschaulich erläutern. Betrachtet man als Gesamtanzahl n der Spielkarten vorerst einmal nur Dreierpotenzen 3^a so ist einleuchtend, dass a (gleichbedeutend mit $\log_3 n$) die Anzahl der maximal notwendigen Durchläufe ist, um die gewählte Karte zu identifizieren. Ein Komplexitätsvergleich zwischen binärer und ternärer Suche sowie die Diskussion, welches dieser Verfahren bevorzugt werden sollte kann abschließend in die allgemeine Einteilung von derartigen „Teile und Herrsche"-Suchverfahren in die Komplexitätsklasse O(log n) münden.

Übersinnliche Kräfte

Beim im Folgenden beschriebenen Trick kann nicht nur der Trickablauf selbst als Algorithmus dargestellt, sondern insbesondere auch die Korrektheit des Algorithmus formal bewiesen werden.

Jeweils die Karten As bis 5 zweier Spielkartenfarben werde aufsteigend hintereinander in einen Stapel sortiert (siehe Bild 3). Der Mentalist behauptet die nun folgenden Schritte der mitwirkenden Zuschauer mental beeinflussen zu können und ihr Vorgehen zu kontrollieren. Ein Zuschauer darf nun beliebig oft Abheben. Anschließend werden von oben fünf Karten des Stapels abgezählt, so dass zwei gleich große Stapel entstehen. Jetzt darf ein Zuschauer von einem der beiden Stapel die oberste Karte nehmen und ohne sie zu betrachten an die unterste Stelle dieses Stapels legen. Dieser Vorgang wird dann noch dreimal wiederholt, wobei der Zuschauer jedes Mal die freie Wahl hat, welchen Stapel verwendet. Nun werden die beiden jetzt oben auf den Stapeln liegenden Karten verdeckt als Pärchen zur Seite gelegt. Bei den verbleibenden Stapeln mit jeweils vier Karten führt jetzt ein Zuschauer insgesamt drei wie oben definierte Bewegungen aus und die oben liegenden Karten werden als Paar entfernt. Durch weitere Wiederholungen dieses Verfahrens verbleibt in jedem Stapel schließlich nur jeweils eine Karte. Diese beiden Karten werden nun aufgedeckt – sie besitzen den gleichen Wert. Und auch alle anderen zur Seite gelegten Kartenpaare sind jeweils wertgleich. – Hat der Magier die Zuschauer wirklich manipuliert?

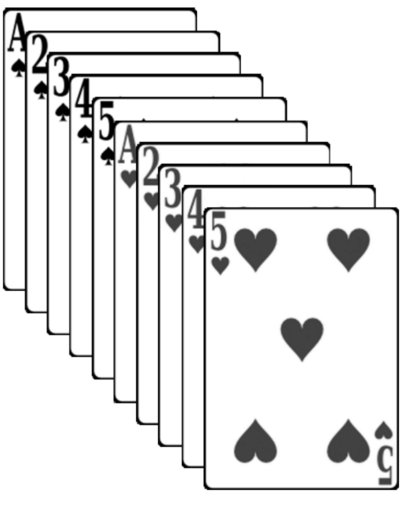

Bild 3: Ausgangssituation (sortierte Liste)

Eine genaue formalisierte Betrachtung des gesamten Trickablaufs legt offen, dass es sich hierbei um einen Algorithmus handelt, der immer zum gleichen Ergebnis (die Karten aller Paare besitzen jeweils den gleichen Wert) führt.

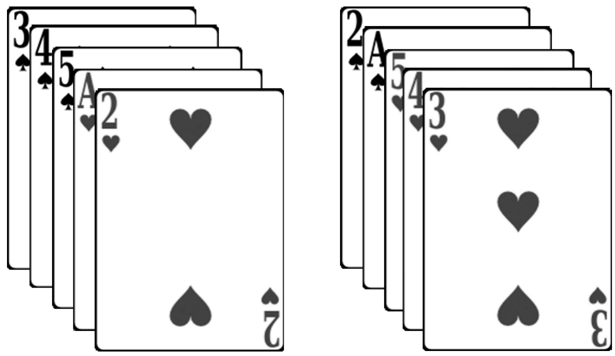

Bild 4: nach der Trennung in Teilstapel

Während beim ersten in diesem Artikel beschriebenen Trick noch eine einfach verkettete (sortierte) Liste als zugrundeliegende Datenstruktur dient, ist hier die Modellierung mit einer *Ringliste* hilfreich. Der Zeiger des letzten Listenelementes zeigt nun nicht auf NULL, sondern zurück auf das erste Element in der Liste. Das beliebig häufige Abheben zu Beginn der Vorführung bedeutet innerhalb der Ringliste lediglich eine Änderung des „Startelements" („Aktuell-Zeigers"), die weitere zyklische Reihenfolge der Elemente wird nicht geändert. Die Halbierung des Kartenstapels in zwei Teilstapel führt dazu, dass sich in jedem jeweils alle Werte von As bis 5 befinden. Dadurch, dass die ersten fünf Karten von oben abgezählt werden, ist die Reihenfolge der Kartenwerte im ersten Teilstapel invertiert gegenüber der im zweiten (siehe Bild 4). Die Einträge in den zugehörigen Teilringlisten lauten somit:

Liste1 = [Wert(1), Wert(2), … , Wert(n)]

Liste2 = [Wert(n), Wert(n-1), … , Wert(1)]

Der „Aktuellzeiger" steht zu diesem Zeitpunkt jeweils auf dem ersten Eintrag in obiger Darstellung, also auf Wert(1) in Liste1 und Wert(n) in Liste2. Alle nun folgenden Zuschaueraktionen verändern innerhalb der Teilringlisten nur die Position des Aktuellzeigers, die Reihenfolge der Einträge bleibt wiederum unverändert. Bei einer Anzahl von n Karten in jedem der Teilstapel darf der Zuschauer genau n-1 Aktionen durchführen. Diese dürfen beliebig auf beide Stapel aufgeteilt werden. Bezeichnet man die Anzahl der in Stapel 1 ausgeführten Aktionen mit *a*, die in Stapel 2 mit *b*, so muss gelten: $a + b = n - 1$

Löst man diese Gleichung nach *b* auf, so erhält man: $b = n - 1 - a$

Daraus ergibt sich für den Aktuellzeiger in Stapel 1 nach Durchführung aller *a* Aktionen die Position

$e1 = 1 + a$

In Stapel 2 erreicht der Aktuellzeiger nach Durchführung der *b* Aktionen schließlich die Position

$e2 = n - b = n - (n - 1 - a) = n - n + 1 + a = 1 + a$

Damit ist formal bewiesen, dass die am Ende aller Zuschaueraktionen in jedem Durchgang aktuellen Werte (oben auf den Teilstapeln liegenden Karten) jeweils die gleichen sind unabhängig von der Wahl von *a* (in Stapel 1 ausgeführten Aktionen) und von der Gesamtanzahl der Karten in den Teilstapeln.

Der sich an die in Bild 4 dargestellte Situation anschließende weitere Ablauf des Tricks kann mit Hilfe eines Algorithmus, der auch Wiederholungen und Zählvariablen enthält, dargestellt werden:

```
setze Zähler i auf 4
wiederhole bis i gleich 1 ist
    wähle j (0 ≤ j ≤ i)
    k = i - 1 - j
    wiederhole bis j gleich 1 ist
        bewege in Stapel1 1 Karte von oben nach un-
        ten
        reduziere j um 1
    wiederhole bis k gleich 1 ist
        bewege in Stapel2 1 Karte von oben nach un-
        ten
    entferne die obersten Karten beider Stapel als
    Paar
    reduziere i um 1
zeige alle Kartenpaare
```

Eine Implementierung des gesamten Tricks mit Hilfe einer Ringliste als zugrundeliegender Datenstruktur ist somit einfach durchführbar.

Sämtliche Ringlistenoperationen wie (sortiertes) Einfügen, Suchen und Entfernen von Elementen können mit Hilfe von Kartenstapeln und dem Einsortieren, Suchen und Entfernen von Karten mit bestimmten Werten veranschaulicht werden.

Außerhalb-Körper-Wahrnehmungen

Der Mentalist lässt sich die Augen verbinden und steuert den gesamten Ablauf des Tricks rein verbal „aus der Ferne". Ein (oder mehrere) Zuschauer legen untereinander vier Reihen mit jeweils vier Karten, wobei beliebig die Bild- oder Rückseite sichtbar ist (siehe 4x4-Matrix in Bild 5). Der Assistent ergänzt eine weitere Zeile mit offenen und verdeckten Karten. Schließlich vervollständigt er durch eine zusätzliche Spalte das Schema zu einer 5x5-Matrix wie in Bild 5. Ein Zuschauer wählte eine beliebige Karte (in Bild 5 den Pik-Buben) und dreht sie von der Bild- auf die Rückseite oder umgekehrt. Der Mentalist, dem die Augenbinde abgenommen wurde, identifiziert sofort die gerade gedrehte Karte.

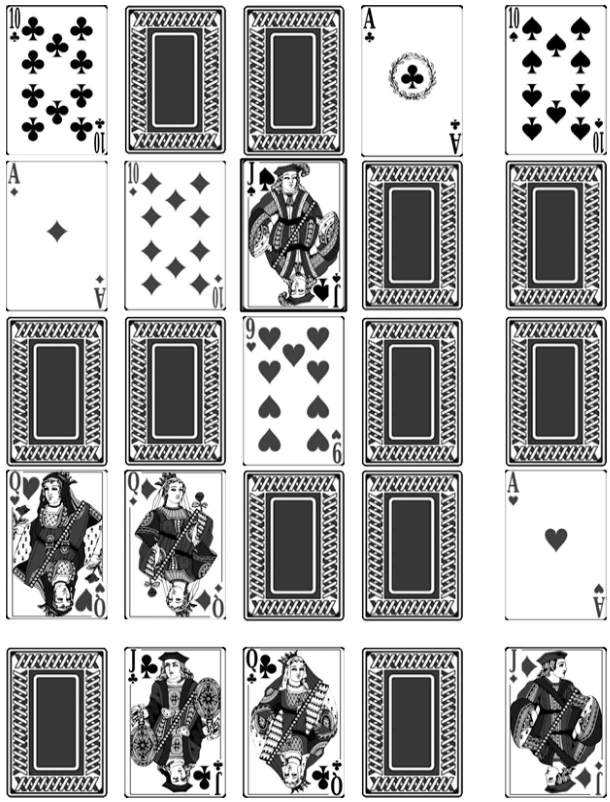

Bild 5: Karten nach Auflegen durch Zuschauer und Assistent

Das informatische Prinzip, auf dem dieser Trick beruht, ist das der *Paritätsprüfung*. Nachrichten in der Informatik bestehen aus einem Strom von gesetzten (1) oder nicht gesetzten (0) Bits. Verwendet man die Null für eine „offene" („Bildseite nach oben") und die Eins für eine „verdeckte" („Rückseite nach oben") Karte, so entspricht die zeilenweise gelesene 4x4-Matrix aus Bild 5 der Nachricht 0110 0001 1101 0011 (siehe Bild 6):

0 1 1 0
0 0 0 1
1 1 0 1
0 0 1 1

Der Assistent legt seine Karten nicht zufällig mit der Bild- oder Rückseite nach oben, sondern setzt seine Karten jeweils dazu ein, jeweils die Parität der Anzahl in jeder Zeile bzw. Spalte offenen (bzw. verdeckten) Karten anzugeben. Hierbei steht eine offene Karte (0) für eine gerade, eine verdeckte Karte (1) für eine ungerade Anzahl. Somit ergibt sich nach seinen Ergänzungen die in Bild 6 dargestellte 5x5-Matrix:

0 1 1 0 0
0 0 0 1 1
1 1 0 1 1
0 0 1 1 0

1 0 0 1 0

Die Störung einer Nachrichtenübertragung bedeutet, dass es durch äußere Einflüsse zu Änderungen der Werte einzelner Bits kommt und der Inhalt der Nachricht somit verfälscht wird. Selbst durch bestmögliche Übertragungsleitungen und hervorragende Abschirmung kann eine absolut störungsfreie Nachrichtenübermittlung nicht gewährleistet werden.

Deshalb wurde z. B. das Prinzip der Paritätsprüfung entwickelt, das mit Hilfe zusätzlicher Prüfbits Fehler, in Nachrichten identifizieren und reparieren kann. Der Zuschauer, der eine beliebige Karte wählt und diese umdreht, stellt die gerade beschriebene Störung von außen dar, die gedrehte Karte ist das verfälschte Bit in der Nachricht.

Die 5x5-Matrix stellt sich nach der Drehung der Karte durch den Zuschauer (Störung der Nachrichtenübermittlung) wie folgt dar:

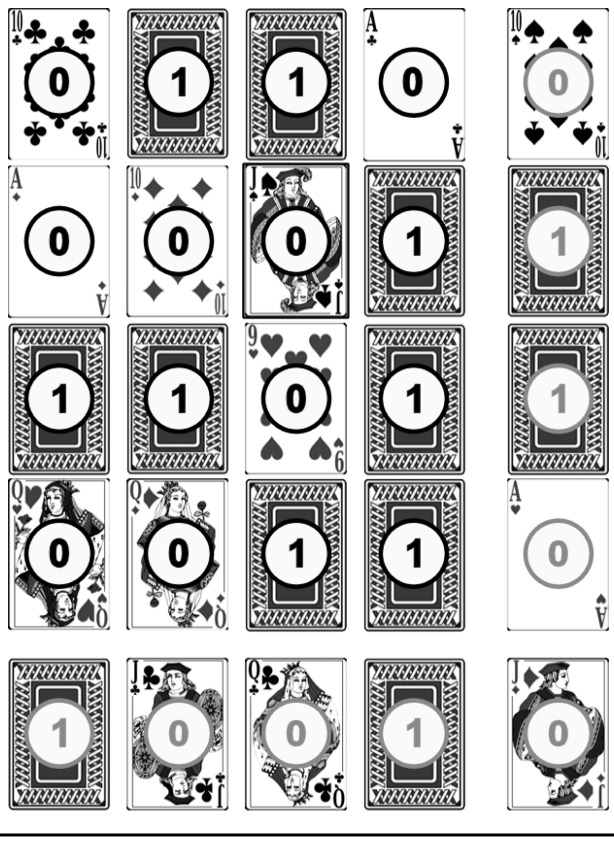

Bild 6: Nachricht (schwarz) mit Paritätsbits (grau)

0 1 1 0 0
0 0 ***1*** 1 **1**
1 1 0 1 1
0 0 1 1 0

1 0 **0** 1 0

Der Mentalist sucht nach Abnehmen der Augenbinde nach der Zeile und der Spalte, in der das Paritätsbit nicht zum Inhalt der Zeile passt und kann somit die gedrehte Karte identifizieren. Wird der gewählte Pik-Bube in Bild 6 umgedreht, so dass seine Rückseite nach oben zeigt, wird aus der bisher dargestellten 0 eine 1. Das Paritätsbit in der zweiten Zeile (1) zeigt an, dass hier eine ungerade Anzahl von verdeckten (bzw. offenen) Karten liegen sollte, was dann nicht mehr der Fall ist. Ebenso passt das Paritätsbit der dritten Spalte (0) dann nicht mehr zur nach dem Umdrehen des Pik-Buben ungeraden Anzahl von verdeckten (bzw. offenen) Karten.

In der Nachrichtenübermittlung wird das gekippte Bit dann nicht nur identifiziert, sondern kann auch gleich repariert werden, denn im Binärsystem gib es nur „richtig“ oder „falsch“ und nichts dazwischen. Das Verfahren funktioniert auch dann, wenn ein Paritätsbit verfälscht wird.

Literatur:

McOwan, P.; Curzon, P.: The Magic of Computer Science: Card Tricks Special or A plethora of pasteboard paradoxes purporting the principles of Computer Science. Department of Computer Science, Queen Mary, University of London

McOwan, P.; Curzon, P.; Black, J.: The Magic of Computer Science II – Now we have your attention... : A medley of magnificently magical marvels mischievously manipulating mind mistakes. School of Electronic Engineering and Computer Science, Queen Mary, University of London