Binary magic cards

Wei Ruen Leong

January 5, 2019

1 Introduction

I was first introduced to the six-card set in Figure 1 by a friend. The 'magic' trick is as follows: I (the participant) am told to pick one number from 0 to 60 without telling my friend what it is. Then, I am instructed to circle the cards that contain the picked number. Suppose that I pick the number 49, then I have to circle the three cards as shown in Figure 1. Based on the cards I have circled, my friend is able to tell me what number I have picked.

How does the magician identify the picked number based on what cards that are circled and not circled? A crude method is to examine the common numbers on the circled cards one by one. The set of common numbers shared among the circled cards in Figure 1 is {49, 51, 53, 55, 57, 59}. For example, the last shared number 59 is also present in the non-circled middle left (green) and bottom right (black) cards. If the picked number is 59, and by the instruction of circling the cards containing the picked number, then the green and black cards have to be circled. Since the green and black cards containing the number 59 are not circled, one can conclude the picked number cannot be 59. Through the process of one-by-one elimination, only the number 49 in the set of common numbers satisfy the condition that only the circled cards contain the number 49, whereas the non-circled cards do not contain that number. Therefore, one can conclude that the picked number must be 49 for the given example in Figure 1.

The crude method is cumbersome to implement if there are many common numbers shared by the circled cards. Fortunately, it turns out there is an easier method of determining the picked number. The magician can tell what is the picked number by summing up the first number in each circled card. For example, summing up the first number in each of the circled cards in Figure 1 yield 49 (1+16+32), therefore confirming the picked number. To the observant readers, it should be immediately noticeable

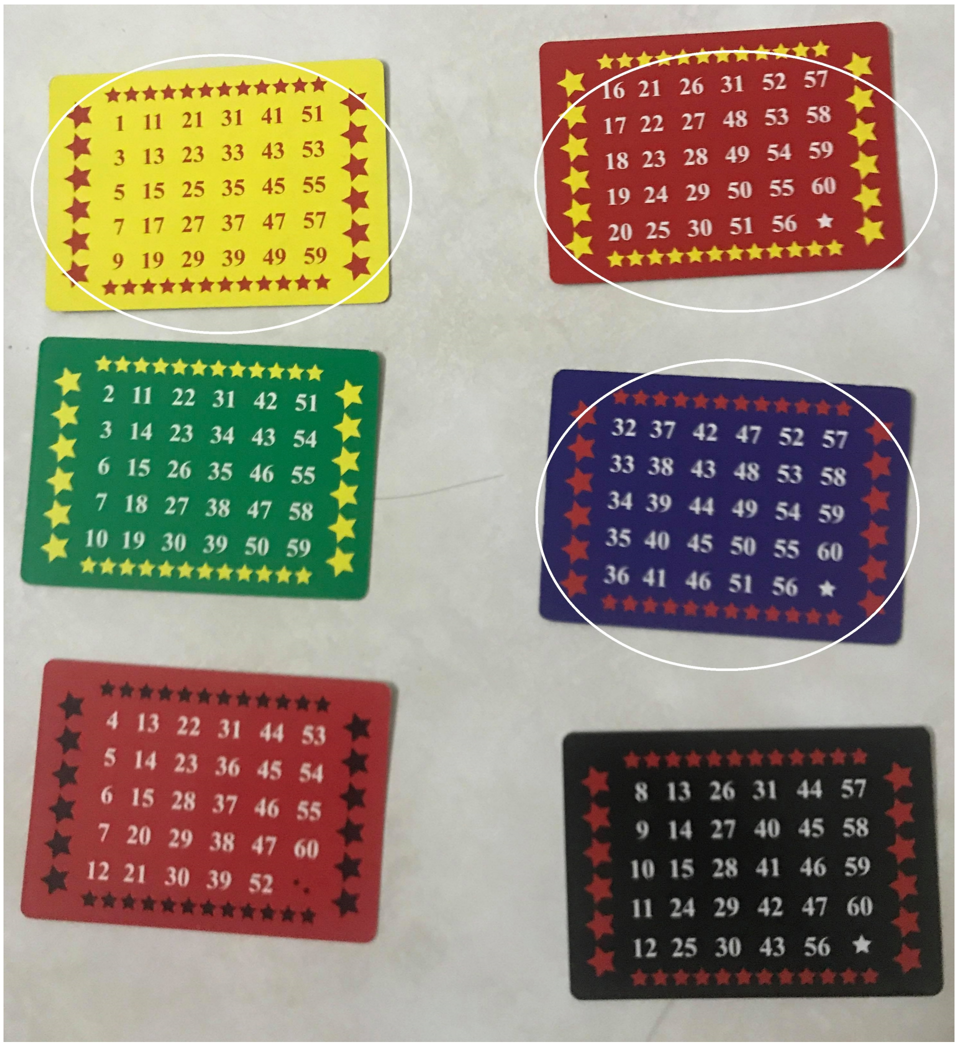

Figure 1: An illustration of the six-card set. Suppose the number chosen is 49. I had to circle the three cards as shown above because they contain the number 49.

that the first number in each card is given by the following sequence of numbers: $2^0 = 1$, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, $2^4 = 16$, and $2^5 = 32$. For this reason, I name the cards binary magic cards because the cards are constructed based on the binary number system (base-2 number system). The cards shown in Figure 1 are for the range 0 to 60. In the next section, I explain how can one construct the binary magic card sets given any range that begins with 0, i.e. how many cards are required for the magic trick and what are the numbers to print on each card?

2 Construction of the binary magic cards

Say the magician has a range of 0 to m in mind, where m is a positive integer. Let the number of cards needed to print out given by n. To determine n, one has to count the number of digits in the binary number representation of m. Let m be 60 as given by the example in Figure 1. If m is 60, then 111100 is the binary number representation of 60. The number of digits in 111100 is 6, therefore n is 6, or in other words, 6 cards have to be printed for the magic trick. How can one determine what numbers to print on each of the 6 cards? One has to first find out the binary number representation (with width n) of each number in the pre-defined range starting from the number 1 until m. For the same example with $m = 60$, the binary number representation for each of the number from 1 to 60 is given as follows in the list:

Number (in base 10)	Binary number representation
1	000001
2	000010
3	000011
⋮	⋮
58	111010
59	111011
60	111100

Recall that there are 6 cards to be printed for the given example, each of the binary number representation is therefore given in 6 digits for reasons that will become obvious soon. Take the first number on the list, the binary number representation of the number 1 is 000001. It turns out the number is telling one to print (indicate by 1) or not to print (indicate by 0) in each of the digit position. The mapping of the digits to the card positions is as follows:

6th card	5th card	4th card	3rd card	2nd card	1st card
0	0	0	0	0	1

How can one interpret the binary number for the printing problem? If it is 1, then print on the card on that position; and if it is 0, then do not print on the card in that position. Reading from right to left, the first number is 1, the second number is zero, the third number is also zero and so on. This means that the number 1 has to be printed only on the first card. For the next number on the list, the number 2 has a binary number representation of 000010, this means that the number 2 has to be printed only on the second card. For the next number on the list, the number 3 has a binary representation of 000010, this means the number 3 has to be printed only on the first and second cards.

On a related note, the print or not to print from the magician's construction perspective is related to the to circle or not to circle from the participant's perspective. This is because say a number x is picked by the participant, then the participant can write down the number x in binomial representation, and the cards circled correspond to the 0 (do not circle) and 1 (circle) representations of the printing problem. Therefore, the construction of the binary magic cards is consistent to the magician's instruction of asking the participants to circle the cards containing the picked number.

A computer program can be written to record the numbers to be printed on each card as one is going through the loop of 1 (in ascending order) until m. When the loops ends, the computer program will return the set of numbers to be printed on each card. I had written a basic computer program to demonstrate a implementation of the binary magic card construction in the file *"BinaryMagic.py"*. The program asks the user for the number m and print outs the set of numbers to be printed on each card (along with the length of each set, i.e. the number of numbers to be printed on each card).

If the readers have noticed, I am still missing a number here. What about the number 0? This is because 0 is not printed in any of the card but it is still a valid choice in the range. If the participant picks the number 0, he/she does not need to circle any of the cards. The magician can then infer that the picked number must be 0.

The example with $n = 6$ given in Figure 1 has a total of $2^6 = 64$ different possible binomial number representations. The range 0 to 60 has a total of 61 different numbers, 3 numbers are missing out of the 64 different possibilities. In Figure 1, the design of the cards omit printing the numbers 61 (111101), 62 (111110), and 63 (111111), where the numbers in parentheses are the corresponding binary numbers. One can also ask the following question: Given a positive integer n, without omitting the printing of any number in the range of 0 to m, what is the maximum m that can be constructed for the binary magic card problem? The maximum m can be calculated by the general formula: $2^n - 1$, the minus one is because one starts counting from 0 in the range. To summarize, for an n-card set of binary magic cards, the range that can be constructed is from 0 to m, where $m = 2^n - 1$. With a slight modification to my original program, I had written the program *"BinaryMagicInv.py"*. The program asks the user for the number n and print outs the set of numbers to be printed on each card (along with the length of each set, i.e. the number of numbers to be printed on each card).