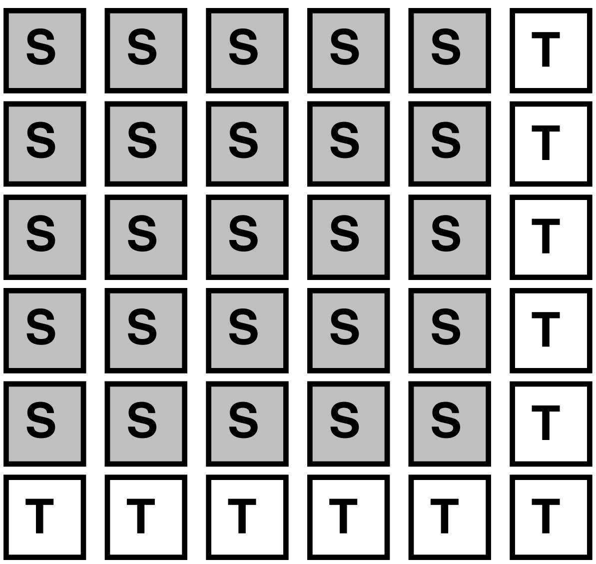
A Bit Of Luck

Preparation.

The student is given 25 cards, one side has a dark face and the other side has a light face. . They are asked to put the cards on the desk to form a 5 by 5 square. They can put as many of the cards facing up as they like and put the rest facing down. After making the square they can decide to flip any of the cards over.

Procedure.

The teacher says that they want to make the trick “just a bit harder” and adds a column of cards on the left end of the square and a row of cards at the bottom of the square. Some of the new cards are face up and some face down. There is now a 6 by 6 square on the desk with some dark faced cards showing and some light faced cards showing in a random pattern selected by the student.



The student lays down 25 cards, some dark face up and some light face up, to form a 5 by 5 square shape.

The teacher lays down 1 additional column on the right side and 1 additional row at the bottom of the square.

You add a card at the right end of each of the 5 rows so that you create an **even number of white faces in each row.**

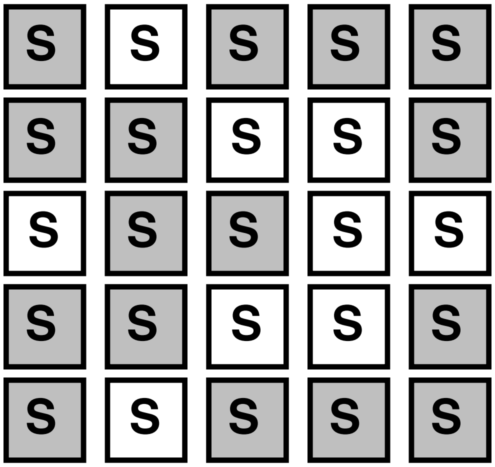
After that you then add a card to the bottom of each of the 6 columns so that you create **an even number of white faces in each column.**

When they are done you turn around and say **“I only have a 1 in 36 chance of guessing which card you turned over. I will need a Bit Of Luck to get it right.”** Look at the cards and then announce the card they turned over.

How it’s done:

The row and column with an odd number of white cards is the location of the card the student flipped over.

Example 1



Row 1 has 1 white student card, you need to add a white card

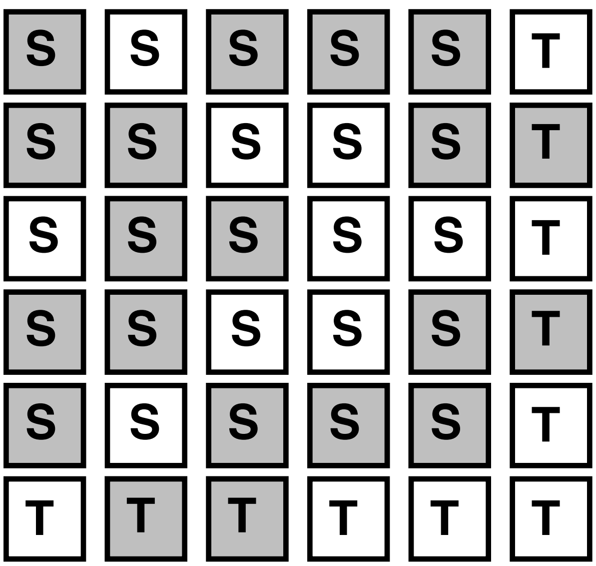
Row 2 has 2 white student cards, you need to add a black card

Row 3 has 3 white student cards, you need to add a white card

Row 4 has 2 white student cards, you need to add a white card

Row 5 has 1 white student card, you need to add a white card

Column 1 has 1 white student card, you need to add a white card
Column 2 has 1 white student card, you need to add a white card
Column 3 has 2 white student cards, you need to add a white card
Column 4 has 3 white student cards, you need to add a white card
Column 5 has 1 white student card, you need to add a white card
The **NEW Column** 6 has 3 white student cards, you need to add a white card



The student lays down 25 cards, some dark face up and some light face up, to form a 5 by 5 square shape.

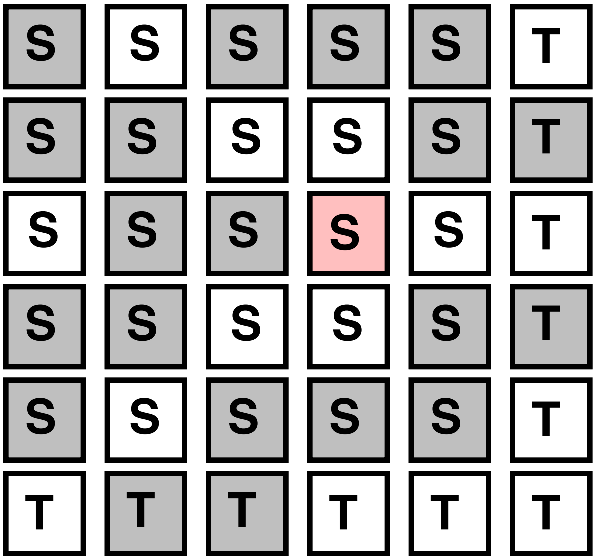
You add a card at the right end of each of the 5 rows so that you create an **even number of white faces in each row.**

After that you then add a card to the bottom of each of the 6 columns so that you create **an even number of white faces in each column.**

Note: Be sure and check that each row and column has an even number of white cards before you proceed.

Example 1

The student picks the **white card in the 3rd row and 4th column** and turns it over so its black. I show it in red so its easier to see.



When the teacher looks at the cards they see that:

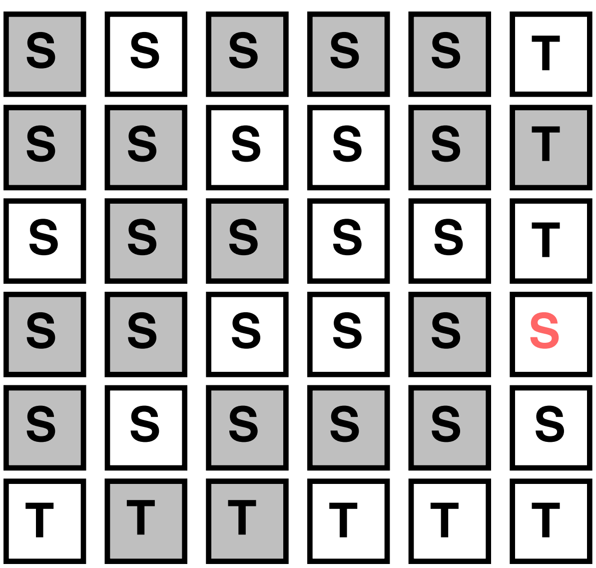
Row 3 has an **odd number of white cards**
and
Column 4 has an **odd number of white cards**

The student turned over the card in the 3rd row and 4th column.

When I announce the card the student turned over I pick it up, flip it over and put it back. This resets the cards so that I can repeat the trick if I want to.

Example 2

The student picks the **black card in the 4th row and 6th column** and turns it over so its white. I made the S red so its easier to see.



When the teacher looks at the cards they see that:

Row 4 has an **odd number of white cards**
and
Column 6 has an **odd number of white cards**

The student turned over the card in the 4th row and 6th column.

When I announce the card the student turned over I pick it up, flip it over and put it back. This resets the cards so that I can repeat the trick if I want to.

Materials:

The easiest front and backed cards I have found is to use playing cards from the Dollar Store (2 decks for $1). The different numbers on the white faces seems to provide a distraction. If you mixed two red and blue backed decks but treated both their backs as dark sides that would provide even more distraction. An added benefit is that you have a name to attach to the card they flipped over. If they flip the back over to the front and the face showing is the 5 of clubs you can name the card and not reference the position at all. If they flip the front of the 5 of clubs over to its back pick up that card show it to the student and say your card is the 5 of clubs. In either case be sure to replace the card in its original position so you can perform the trick again.

If playing cards do not work for you then cut out squares from white paper and dark paper. Use a glue stick to make 36 squares. I keep them in a baggy for use on students desks.

White board demonstration:

Michael's has sheets of magnets that have a dark side and a white side. If you want to spring for the cost they make great white board squares if your white board is the right type.

Ideas to consider:

The trick would work just as well if you required the rows and columns to have an even number of dark cards instead of white.

The trick would work just as well if you required the rows and columns to have an odd number of white cards instead of an even number. We call this even **parity**.

The number of cards the student is given is not limited to 25. I often use a 4 by 4 square and my added row makes a 5 by 5 square. Less then 16 cards makes the trick seem less amazing and more than 36 just takes too long to set up.
Any rectangular arraignment for the cards would also work. The reason for the square is based on the fact that this trick demonstrates a very important concept in computer science and the common lecture uses a square array.

Base 2 and Bits and Bytes

Computers use strings of zeros and ones to represent numbers and letters. Any given number or letter can be expressed as a binary string. The number 31 is represented by the binary string 0 0 0 1 1 1 1 1. The capital letter Z is represented by the binary string as 01011010.

We refer to each number in the string as a digit. There are only two possible digits , zero or one, for each number in the string. For this reason, we refer to each number in the string as a binary digit.

Binary digits are called Bits

A string of 8 zeros and ones is called an **8 bit string**. Each eight bit string is called a **byte.** Think of the bits as letters in a word and the entire 8 bits as one word. This is the language of the computer. They speak in words called bytes. Each word in the computer is eight letters long. Each letter is called a bit. The computer talks in 8 bits words called bytes.

When a computer stores an 8 bit word it also stores one other digit or bit with it. A bit is added to the end of the 8 bit string. This added bit is used to tell if the total of the 8 bits is even or odd. We call this added bit a parity bit or check bit.

Parity Bits

A parity bit, or check bit, is **a bit added to the right end of a binary string** that indicates whether the the total of bits in the string is even or odd. **If the total of the 8 bits is an even number then the parity bit is 0. If the total of the 8 bits is an odd number then the parity bit is 0.**

8 bit binary code	parity	8 bit binary code	parity
1 0 0 0 0 0 0 1	0	1 0 1 0 0 0 0 1	1

The following paragraph was taken from a college computer science lecture

In case of even parity, if the count of ones in an 8 bit string is odd, the parity bit is set to 1. This makes the count of ones in the entire set of 9 bits even. If the count of ones in an 8 bit string is even, the parity bit is set to 0. This makes the count of ones in the entire set of 9 bits even.

But wait, that is what our trick does

Can you see that this is exactly what we did in our trick. We used a 5 bit string instead of an 8 bit string. We added cards to the right end of the rows and and the bottom columns of the student cards to make the number of white cards an even number in each row and each column.

Wow,
when you do this trick with your students
you become an introduction to Computer Science teacher

The cards you add are parity bits

We used the white cards as 1's and black cards as 0's. We then added parity bits to make the parity of each row and column even. Parity bits are used as the simplest form of error detection.

This trick is based on the use of the parity bits. The cards you add at the right of the rows and on the bottom of column are parity buts. If the row has an even number of face of cards you add a face down card. If the row or columns has an odd number of face up cards you add another face up card When you are done you have forced the number of face up cards in each row and column to be even You have created an even parity.

When the student picks any card and flips it over the number of face up cards in the row and column their card is in becomes odd. To find their card you look for the row and column with an odd number of face up cards (an odd parity). That is the location of their card. It's that easy.

Error Detection

Computers send a lot of information to storage devices or to other computers over the internet. The data is sent as strings of 0's and 1's . The data might be changed (corrupted) in the transfer process and create an error in transmission. If you could detect a transmission error then you could require the corrupted part of the data to be retransmitted and ensure the data is correct. The use of parity bits allows the detection of corrupted strings of bits.

If an odd number of bits (including the parity bit) are transmitted incorrectly, the parity bit will be incorrect, thus indicating that a parity error occurred in the transmission. The parity bit is only suitable for detecting errors, it cannot correct any errors. The data must be discarded entirely, and retransmitted. A web search for " hamming code for an example of an error-correcting code.

A "bit" of fun

When the computer scientist wrote **1**   0 they were told to move the 1 over a "bit"
so he wrote 0 **1**

1 0 1 1

why is the zero smaller than the ones

I will tell you in a little bit

The farmer said was amazed at the large increase in the number of rabbits he had.
The number of rabbits had increased from 11 to 111
His computer science friend replied that's just a "bit" of an increase

Checkout Scanners and Bar Codes

Most barcodes are 10 digits long. The first 9 digits identify the product and the last digit is the checksum. $d_9 \quad d_8 \quad d_7 \quad d_6 \quad d_5 \quad d_4 \quad d_3 \quad d_2 \quad d_1 \quad \text{cksum}$

A bit added at the end of a bar code acts like a parity bit. It is called a check bit but is also called a checksum character. The purpose of a check bit is to verify that the information on the barcode has been entered correctly. The barcode reader's decoder calculates the checksum by performing a series of mathematical operations on the digits that precede the check bit, and compares the result of the calculation to the value of the check bit. If the check bit matches the result of the calculation, the reader emits a beep to acknowledge that the results match, and the scan has been successful.

The following proceeder is used to determine if the code has been scanned correctly.

$$\text{MOD } 10 \ \left[3(d_9 + d_7 + d_5 + d_3 + d_1) + (d_8 + d_6 + d_4 + d_2)\right] = \text{cksum}$$

Universal Price Code

The 10 digits of the UCP are listed under the bar code. The first 9 digits identify the product and the last digit is the checksum. $d_9 \quad d_8 \quad d_7 \quad d_6 \quad d_5 \quad d_4 \quad d_3 \quad d_2 \quad d_1 \quad \text{cksum}$

1. Find the sum of the 6 digits in the odd positions and multiply by 3.
2. Find the sum of 5 the digits in the even positions.
3. Add the two sums.
4. Divide the total by 10. If the remainder is 0 that is the check sum. If the remainder is not zero subtract the remainder form 10 and that is your check sum.

The scanner scans the bar code and calculates the checksum. If is the one listed the scanner accepts the scan and records the purchase.

The UPC on the back of my Altions Peppermint The Curiously Strong Breath Mint have a UPC of

2 **2** 0 **0** 0 **1** 5 **9** 3 **3** - 5

Calculate the check sum

1. Find the sum of the 5 digits in the odd positions (bolded) and multiply by 3
   2 + 0 + 1 + 9 + 3 = 15 15(3) = 45
2. Find the sum of 5 the digits in the even positions 2 + 0 + 0 + 5 + 3 = 10
3. Add the two sums 45 + 10 = 55
4. Divid the total by 10. If the remained is 0 that is the check sum. If the remainder is not zero subtract the remainder form 10 and that is your check sum.

The remainder when we divide 55 by 10 is 5. That is our check sum

The scanner scans the bar code and calculates the checksum. If it is 5 the the scanner accepts the scan and records the purchase.

ISBN 13

All books for sale have an ISBN number. In use since 2007, ISBN 13 has 13 digits and a check sum digit to the right. of the 13 digits. Its check digit is generated with a similar algorithm used with the Universal Price Code for food products.

The ISBN 13 code for my Pre algebra Math Textbook is 9 7 8 8 0 9 8 4 2 9 5 9 – 6

ISBN: 978-0-9842959-0-6

90000

9 780984 295906

Check to see if the 0 is the correct check sum. 9 7 8 0 9 8 4 2 9 5 9 0

1. Find the sum of the 6 digits in the odd positions 9 + 8 + 9 + 4 + 9 + 9 = 48

2. Find the sum of 5 the digits in the even positions and multiply by 3. 7 + 0 + 8 + 2 + 5 = 22

3. Add the two sums 48 +22 = 60

4. Divid the total by 10. If the remained is 0 that is the check sum. If the remainder is not zero subtract the remainder r form 10 and that is your check sum.

The remainder when we divide 60 by 10 is 0. That is our check sum

The scanner scans the bar code and calculate the checksum. If is the one listed the scanner accepts the scan and records the purchase.