Introduction
Today

• Cryptography
  ▪ Networks and Parity
  ▪ Ciphers
  ▪ Encryption

• Break:
  ▪ 18:30(ish) – 15 Minute Break
Why Cryptography?

GCSE Syllabus outlines:

- Security and
- Encryption

Also fits well with data representation and networks.
Networks and Parity
Overall Idea

Two very common things to do with data are:

- Store it, and
- Send it somewhere (over a network)

During this process, it is very easy for a bit to accidentally change, for example:

- Storage media may be subjected to physical shock
- A network may be faulty

How can we stop these errors?
How Can Data Be Corrupted?

As we have seen, everything is encoded in bits. Hence:

A 1 bit can be changed to a 0 bit, and

a 0 bit can be changed to a 1 bit
Redundancy

The first solution would be to:

- Store the data 3 times
- Send each bit over the network 3 times

Then compare the three copies and look for an error.

Immediate problem -- Wasting space and network usage.

Also -- what if the same bit gets “flipped” in two copies?
A Better Solution: Parity

Parity allows:
1. The detection of errors.
2. The correction of these errors.

Idea: Add some extra bits that encode the state of the data

How to teach this: Through a card trick!
Card Trick Demo
How It Works

Notice we added an extra row of cards -- making the number facing down even!
Detecting The Change

- Flipped card in this row
- Parity row
- Flipped card is in this column
- Parity column
An Example With Bits

Let's say we want to add parity for the following:

001101110010111001

First split it into (for example) 6 bit blocks:

001101
110010
111001

Next add the parity bits.
Adding Parity Bits

Add a bit to each row making the number of 1’s even:

0011011
1100101
1110010

Then transmit the full 21 bit sequence:

001101111001011110010
At The Other End

Split the sequence into block of 7 and check there is an even number of ones:

0011011
1100101
1110010

This will detect errors, but do we know which bit has been changed?
Parity Tasks

1. Split into groups and perform the card trick (2 magicians and 2 volunteers).

2. Add parity bits to the following:
   101111010101010001100
   Hint: Think about the split size.

3. How can we extend the bit parity scheme to allow for correction of errors?
Allowing for Error Correction

Add a bit to each column as well as each row, making the number of 1’s even:

```
 0011011
 1100101
 1110010
 0001100
```
Correcting The Error

Now the receiver can check rows and columns, the flipped bit will be highlighted by incorrect values in both:

```
0011011
1000101
1110010
0001100
```

Such a scheme allows the correction of any odd number of errors.
Parity Tasks (2)

1. Using a 5x5 grid of bits, add parity rows and columns.

2. Now try to change the grid so that the changes are not detectable!
Parity Tasks (2): Solutions

1. Using a 5x5 grid of bits, add parity rows and columns.

   | 0 | 0 | 1 | 0 | 1 | 0  
   | 1 | 0 | 1 | 0 | 0 | 0  
   | 0 | 1 | 1 | 0 | 1 | 1  
   | 1 | 1 | 1 | 0 | 0 | 1  
   | 1 | 0 | 0 | 1 | 1 | 1  

   (parity columns)

   | 1 | 0 | 0 | 1 | 1 | 1  

2. Now try to change the grid so that the changes are not detectable!

   | 0 | 0 | 1 | 0 | 1 | 0  
   | 1 | 0 | 1 | 0 | 0 | 0  
   | 0 | 1 | 1 | 1 | 1 | 1  
   | 1 | 1 | 1 | 0 | 0 | 1  
   | 1 | 1 | 0 | 0 | 1 | 1  
   | 1 | 0 | 0 | 1 | 1 | 1  

   (parity columns)
Ciphers
Peeking Eyes

So far we have looked into error detection and correction, but what if the information we wish to store/send is sensitive?

Examples:
1. Bank data!
2. Personal information (Name, address, email).
3. ...

For this, we will use cryptography to “obscure” the data.
The Setup
Some Terminology

Cryptography -- Crypto (Hidden), Graphy (Writing).

Encryption -- The process of “hiding” data.

Decryption -- The process of “unhiding” data.

Plain Text -- The original message.

Cipher Text -- The encrypted message.
Julius Caesar's Cipher

A very well known form of encryption is the shift cipher.

Idea: Shift each character 3 positions.

Encryption:

    Clear...: ABCDEFGHIJKLMNOPQRSTUVWXYZ
    Cipher.: DEFGHIJKLMNOPQRSTUVWXYZABC

Decryption: Easy its just the reverse :)
A General Shift Cipher

It is easy to generalise the shift cipher.

For example, we can shift by any number:

Algorithm: shift the alphabet n spaces to the right.
Key: the value of n

Note: The key is all we need for decryption!
Caesar's Cipher Tasks

1. Choose a message and encrypt it using the key n=6.

2. Decrypt the following using the key n = 3:
   FHQWXULRQ KLSSRSRWDPPXV WR DWWDYN
   JDXOLVK YLOODJH DW GDZQ

3. How hard is it to “crack” the cipher?
Frequency Analysis

These kind of ciphers are open to frequency analysis on the data, making them fairly easy to attack.
Transposition Ciphers

Another form of cipher is to not “encode” the characters, but to simply re-arrange them.

For example: write out a message in a zig-zag over two lines.

Plain text: THISISSECRETSTUFFWOW

```
T I I S C E S U F O
H S S E R T T F W W
```

Now send: TIISCESUFOHSSEERTTFFWW
Column Transposition Cipher

It is also possible to use column based encryption:

Plain Text: ALLOURDUCKSHAVEFLOWNTOVENICE
Key: 7

Column representation:

SEND IN COLUMNS: AUEOLCFVLKLEOSONUHWIRANCVDVTE
Decoding

To decode, simply reconstruct the columns and read off the sentence :)
Keyword Transposition Cipher

It is also possible to use a word as the key.

Plain Text: ALLOURDUCKSHAVEFLOWNTOVENICE
Key: CIPHERS

Column representation:

<table>
<thead>
<tr>
<th></th>
<th>CIPHERS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>--------</td>
</tr>
<tr>
<td>ALLOURD</td>
<td></td>
</tr>
<tr>
<td>UCKSHAV</td>
<td></td>
</tr>
<tr>
<td>EFLOWNT</td>
<td></td>
</tr>
<tr>
<td>OVENICE</td>
<td></td>
</tr>
</tbody>
</table>

Send in columns alphabetically (CEHIPRS):

AUEOUHWIOSONLCFVLKLERICANCDVTE
Transposition Tasks

1. Encode the sentence:
   INEEDMORETROOPS
   using column based transposition and a key of 5.

2. Decode:
   EEFEA_SISENGEPCIBIICXCHEHRFTNTITOORS_____NEE_L_N_E
   Using a key of RFNEYMA.
   Hint: the spaces have been encoded too!
Modern Encryption Schemes
Principles for Encryption Algorithms

“A cryptosystem should be secure even if everything about the system, except the key, is public knowledge.”

That is: All of the secrecy lies inside the key, and not by obscuring the algorithm!
Symmetric Encryption

Like we have seen so far:

Asymmetric Encryption

Like we have seen so far:

Some Things To Think About

1. How do Alice and Bob share their key for symmetric encryption?
2. Can you decode the following:

   QFL HCVPS PX V ANSWLCEZK NCJVS; PQ XQVCQX QFL BPSZQL RNZ JLQ ZT PS QFL BNCSPSJ VSW WNLX SNQ XQNT ZSQPK RNZ JLQ QN DKVXX

   Hint: use frequency analysis and try common words.
Can you decode the following:

QFL HCVPS PX V ANSWLCEZK NCJVS; PQ XQVCQX QFL BPSZQL RNZ JLQ ZT PS QFL BNCSPSJ VSW WNLX SNQ XQNT ZSQPK RNZ JLQ QN DKVXX

Hint: use frequency analysis and try common words.

Result:

THE BRAIN IS A WONDERFUL ORGAN; IT STARTS THE MINUTE YOU GET UP IN THE MORNING AND DOES NOT STOP UNTIL YOU GET TO CLASS
Conclusion

- Cryptography
  - Parity (Card Tricks and Bits)
  - Ciphers (Shift, Transposition)
  - Encryption (Shared Key, Public Key etc)

- Next Week: Problem Solving