CODES

Concealing Information

Revealing Information

Detecting errors

Correcting errors

Follow-up session

November/December 2008

Instructor's notes for Codes

Section 0 – What are Codes ?

1. Ask the participants what the phrase "code" means to them, have them freely associate phrases in which the word "code" appears, and make a list of meanings on a blank TSP.
2. Incorporate their thoughts into the title page TSP #0 which list four topics that we will touch on today: "concealing and revealing information, error detection, and error correction."
3. Discuss the first two as two separate strands and also about concealing and revealing information at the same time – e.g. transmitting credit card information over the web.
4. In the past decade, there has been a major shift from codes for national security (NSA biggest employer of mathematicians) to codes for economic security.

Section 1a – Substitution codes – when you know the key

The first activity involves substitution codes. In substitution codes, a person encodes a secret message by replacing each character with a substitute in some regular, consistent way. That person gives the encoded message to another person who has the key to the code, and that person then decodes the message. For example (see TSP #1), you could encode the message "I LOVE YOU" by replacing each letter by the letter that comes before it; you would get "H KNUD XNT;" your boyfriend could then decode the message by replacing each letter by the letter that comes after it. (If he didn't know the key, he might conclude that you love Ron!)

Now have the participants try the examples on Hand-out 1 (modified) – the answers are
1. Easy code.
2. Could your students decode this? What about "a"?
3. Euler circuit

Section 1b – Substitution codes – when you don't know the key

Take, for example, the following "cryptoquote" taken from the ***Tampa Tribune***, as shown on TSP #2. If you are given the key, then it is easy to solve. If you are not given the key, then it will be more difficult to solve.

Tampa Tribune April 7, 1995 YTDWTXTKB WY LUF YDWFZDF PWLU LUF KAFRLFYL ZSEIFA TG EFLUTNY RZN LUF XFRYL AFYSXLY.-----QTWZDRAF	***Tampa Tribune*** April 7, 1995 SOCIOLOGY IS THE SCIENCE WITH THE GREATEST NUMBER OF METHODS AND THE LEAST RESULTS.-----POINCARE

However, the fact that cryptoquotes appears in almost every paper every day implies that lots of people are able to solve them. This illustrates that secret codes which are "substitution codes" are the simplest sort of codes and secret messages that are sent using substitution codes are easily

broken.

Let's see how we can try to break the code on another secret message – a cryptopoem. Show them TSP #3 which has an encoded version of "The Raven." Don't tell them what this is (in fact, you never will) but let them brainstorm, with you writing ideas on the board, about ways to decode this message.

One method of deciphering substitution codes containing English text is to consider the frequency with which the various letters appear in the encoded message. The frequency with which each letter appears in English writing is well-known (see TSP #6 reproduced at right), and can be compared with the frequency with which the encoded symbols occur. Can you tell which symbol (in wingdings font) represents the E? The T?

After a few minutes, give them Hand-out #2 (same as TSP #4), which has the same quote, encoded now with letters instead of the more complex wingding symbols. Ask each group to work together to determine how many times each letter appears in the cryptopoem and record the results on TSP #5. Of course, a longer message would likely have a distribution more like that indicated by the chart above, but it is still somewhat effective for short messages as well.

Order and Frequency of Single Letters

E 12.31%	L 4.03%	B 1.62%
T 9.59	D 3.65	G 1.61
A 8.05	C 3.20	V 0.93
O 7.94	U 3.10	K 0.52
N 7.19	P 2.29	Q 0.20
I 7.18	F 2.28	X 0.20
S 6.59	M 2.25	J 0.10
R 6.03	W 2.03	Z 0.09

Give them a few minutes to work on decoding the cryptopoem; those tables that don't complete it will have an opportunity to finish decoding it later in the morning.

If it seems appropriate, show them TSP #7 which shows the frequency of pairs and triples that occur together very often. Don't dwell too long on this, however.

Section 1c – Substitution codes – when you want to convey information

While substitution codes are good for concealing secret messages from those with whom we wish not to share our information while, at the same time, conveying information to those we want to tell something, they are used more often for conveying information to everyone in situations when written text doesn't suffice. Can the participants think of any codes which are used frequently? For example, ASCII, semaphore, braille, Morse Code, etc. TSP #8 (semaphore and Braille) and TSP #9 (Morse code and phonetic) shows some of these.

Section 1d – Substitution codes – when you want to both convey and conceal information

In military and diplomatic situations, substitution codes are used both to conceal and to convey information – you want to get the information to your allies and you want to conceal the information from your enemies. Even Julius Caesar used simple substitution codes. But they are pretty easy to figure out, particularly if the message is a long one. If you want to use a substitution code and make it hard to read the message, one way to do that is to delete the spaces

between words and group the letters into five-letter "words." TSP #10 shows a famous example – the Zimmerman telegram, sent by the German Foreign Secretary, Arthur Zimmermann, on January 16, 1917, to the German ambassador in the United States of America, Johann von Bernstorff, at the height of World War I. On January 19, Bernstorff, per Zimmermann's request, forwarded the Telegram to the German ambassador in Mexico, Heinrich von Eckardt. The Telegram instructed Ambassador Eckardt that if the United States appeared likely to enter the war he was to approach the Mexican government with a proposal for military alliance. He was to offer to help Mexico reclaim Texas, New Mexico, and Arizona that it had lost in the Mexican-American War. The Zimmermann Telegram was intercepted and decoded by the British cryptographers. The revelation of its contents in the American press on March 1 caused public outrage that contributed to the United States' declaration of war against Germany on April 6.

In World War II, Germany created a sophisticated machine, called Enigma, that it used to create keys to encrypt and convey messages to commanders in the field. British cryptographers cracked Enigma and were able to intercept secret German messages, and it is estimated that the war would have lasted two years longer if they had not done so. An interesting bit of history. The Germans carried out a number of bombing raids on British cities, the most devastating of which was the attack on Coventry in November 1940. It is claimed that the British did not prepare for that attack even though they knew it was coming because they didn't want the Germans to know that they had cracked Enigma. This claim is unsubstantiated.

Nowadays, when we buy something online, we want to send a message to a company that enables them to charge our account $100, but we don't want anyone else to be able to read or use that message – again, we want to convey a message to an ally and conceal the message from a foe. The examples we have looked at so far all involve a "secret key" that only you and your ally knows. Thirty years ago, the idea of a "public key" was invented and that made possible all of the transactions that we conduct today. Explain using TSP #11-13 (all on Handout #3) the difference between "secret key" and "public key" and give example of how they are used. Note: A brief review of prime numbers is appropriate here.

THE FOLLOWING IS REPLACED BY THE DISCUSSION OF TSP #11-13. This is just the tip of the iceberg when it comes to using numbers to encode information. Since computers are very fast with numbers, it is possible to have them do very complex manipulations, far beyond the scope of simple substitution codes, which result in essentially unbreakable codes. For example, the RSA encryption system, also called "public-key encryption", is a coding scheme where an individual or corporation (let's use a bank, for example) can publish what is called a "public-key". A public key is a huge number of about 230 digits which is the product of two huge primes of about 115 digits each. Using this public key, anyone can encode a message and send it to the bank, but only someone who knows the prime factorization of the public key, namely the bank, is able to decode the message. Anyone else who can factor that 230 digit number would be able to read the encoded messages also, but this is one of the biggest unsolved problems in mathematics...finding a method for quickly factoring large numbers, or proving that no quick method exists.

Section 2 – Error Detection

As we saw, not all substitution codes are for keeping information secret, but rather for more efficient transmission of information. For example, bar codes in supermarkets, bar codes on library books, bar codes on mail, ISBN numbers, drivers license numbers, credit card numbers and so on are all used to transmit information. If a scanner in a store scans an item for its bar code but, due to a smudge or something reads the wrong code, it would be good if there was some way for the computer to detect that it hadn't read the code correctly so that it can try again.

On your tables you will find examples of cards on which postal codes are printed. What do you notice about the postal codes – all consist of long and short bars, all have the same number (52) of bars, and all start and end with a long bar. Actually, the bars are in groups of five. We are going to see why that is the case and how the ZIP+4 codes are encoded.

Summarize using TSP #14 and encourage the participants to find all (10) ways to arrange 2 long and 3 short bars in a row, as the two examples indicate. Work through this with them on the board, and when you have all 10, have someone systematically list them (if you haven't already). Connect this with "5 choose 2." Then move to TSP #15 and show how the numbers from 1 to 9 and 0 are encoded with the alphabetical list of 10 configurations of 2 long bars and 3 short bars.

We could use TSP #15 to decode zip codes, but having to look up the groups of 5 in the table takes too long. TSP #16 provides a shortcut. Use the shortcut to decode the encoded ZIP+4 code on TSP #14. Then have them use the shortcut on TSP #16 to decode the encoded ZIP+4 code on one of the cards on their table.

At the end, however, there seems to be an extra group of 5 bars, i.e., an extra digit. Anyone know what that is? It's a "check digit"! Explain that in order to help insure that a scanner has correctly read the bar code for the ZIP+4, the postal service adds a "check digit" at the end. (See TSP #17.)

How is the check digit determined? Ask each person to add all ten digits on one of the postcards and ask the people in each group to discuss what their sums have in common.

You see, to send 9 postal digits, the post office uses 10 groups of 5 bars, representing 10 digits. The last digit is determined to be that one which will make the sum of the 10 digits a multiple of 10. If after reading the digits from an envelope the computer had determined the sum to be, say, 41 or 39, then since this was not a multiple of 10, we can be sure that it had decoded something wrong and it had better try again. This is the purpose of "check digits." It helps the computer recognize that the scanner had picked up a wrong digit, or that the code had been printed wrong, and so the piece of mail must be sorted by hand..

Now have them work on the problems on Handout #4; review them using TSP #18.

Another good feature of the postal code is that if the scanner scans a single bar wrong (or if a single bar got printed wrong) then the computer can pick that up, too, because there won't be the right number of longs and shorts in a group. That is the point of Problem 3a. In this case the problem can be corrected.

But the scanner can't correct the error in Problem 3b because one of the numbers was coded

incorrectly, and you can't tell which one it was.

Summarize the "error detection and correction" capabilities of bar codes using TSP #19.

OMIT: What are some weaknesses of the postal code? Well, if the scanner picks up two wrong digits, then it's possible (though unlikely) that one error will cancel the other and you'll still get a multiple of 10. But the chances of this are VERY small.

Another good example of an error-detecting code is the ISBN system of book coding (see TSP #20). Each book has a unique ISBN (international standard book number) which can usually be found in the front of the book. Here's how it works: Each ISBN is 10 digits long. The first group of digits is the language of the book, the second group is the number of publisher, the third is the number of the book, and the fourth is the check digit. The groups can have different sizes.

How does the check digit work? A dramatic way to demonstrate this is to ask someone to produce a book and tell you the first nine digits of the ISBN. You will tell them the last digit! You do this by multiplying the given sequence by 10, 9, 8, 7, 6, 5, 4, 3, 2 and adding. The last digit is what you would need to add to this sum to make it a multiple of 11. Note, if it's a 10, then the digit is 'x'.

This is discussed in TSP #20-22. If you multiply the digits in the ISBN by the pattern (10, 9, 8, 7, 6, 5, 4, 3, 2, 1) and add them together, you should get a multiple of 11.

Now have them do the problems on Handout #3, and discuss problem #2 using TSP #23.

Stress that the main error in entering ISBNs is that two digits of the ISBN are transposed, not that the digits are entered incorrectly. So if you have an incorrect ISBN, like 0-7168-1186-9, you can consider what transposition was most likely. Use TSP #23 and TSP #24 to assert that whereas there were eight possible ISBNs which would have resulted in 0-7168-1186-9 if a typing error were made, there is only one ISBN which would have resulted in 0-7168-1186-9 if a transposition were at fault. So although the check-digit calculation is more complication for ISBNs than for ZIP codes, ISBN is pretty good at correcting transposition errors. We will see how this works in the last handout, later in the workshop.

This is done in Problem #3 in Handout #9. When reviewing that problem, point out that if you transpose two numbers whose difference is 4, that causes the total to change by 4 – but when you move the larger number to the left the total increases by 4 and when you move the larger number to the right the total decreases by 4. So you can figure out that the most likely error in the incorrect ISBN, 0-7176-1186-9, is a transposition. Since the sum 210 is 1 more than a multiple of 11, the most likely possibility is that two adjacent digits that differ by 1 were transposed. In this case, it was the 7 and 6, which differ by 1, that were transposed – so the ISBN should have been 0-7167-1186-9.

To explain why this happens, write the following on the board (this appears in problem #3 on Handout #9): If you reverse two adjacent numbers LR in the ISBN code, so that what appears is RL, the following happens – if R is larger, then R-L is added to the sum, whereas if L is larger, L-R is subtracted from the sum. If there is time, explain algebraically why this is correct. The

result is that you can often find a transposition error by this method.

Section 3 – Error Correcting Codes

When data is transmitted from a satellite to earth, for example, or when data is read from a CD or transmitted by modem, it is sent as a sequence of 0's and 1's. It is useful for the receiving device to be able to check to make sure the data was transmitted correctly (despite interference or static that can intrude on the way.) If a long string of 0s and 1s is sent, and a few of them get changed in transit or read incorrectly, then the checksum will find it. But at that point it would be nice to be able to correct the problem without having to ask the CD or satellite to retransmit the data. Codes which allow this to be done, assuming limited data corruption, are called "error-correcting codes."

Distribute Handout #6 and lead the participants through the exercise, also shown on TSP #25. Some problems with this non-error-correcting code are that there are words which differ by only a single letter (for example CAT- SAT, GOT-NOT, HIP-ZIP-LIP, YET-JET, etc...). Thus, if the 'H' in HIP were changed to a 'Z' we might think it was a code for 'Z', and we wouldn't even detect an error...and the message 'ZELP' would come through, instead of 'HELP' as on the slide. Then have them try to find words so that no two words have two letters in common. This looks better...now errors can be detected. Can they be corrected, though? No! Have them say why.

Take MOP and NOT for example. They differ by two letters, so a single error can't change one of them to the other, and hence a single-letter error can be detected. But it can't be corrected because, if in the code word MOP, the P got changed to a T, we would receive the code word MOT. It is then unclear which word was intended here. It might have been a MOP with the P changed to a T, or a NOT with the N changed to an M. So, have them articulate that if this type of error-correction was to be successful, we would need any two code words to differ by at least three letters. Discuss the Abel, Baker, Charlie code on an earlier TSP. Discuss this conclusion using TSP #25 and how the three-letter code that maps each letter into three of that letter (e.g., A becomes AAA) might work.

This is the idea behind most error-correcting codes that are used in actual data transmission. Long strings of 0s and 1s are encoded somehow and sent in such a way that, even if there are modest errors in transmission, the receiver can decode the message and make the minor corrections.

A very simple type of error-correcting code is shown on TSP #26. In this example, a zero is replaced by 000 and a 1 is replaced by 111. A single error can thus be detected and corrected. Any word with a single 1 is assumed to be 000 and any word with two 1s is assumed to be 111. Have the participants decode the message on Handout #7, assuming no more than one error per block of 3.

That's as far as I got – that is, we showed that there is a simple error-correcting code. I asked why we didn't triple the zip code and many correctly realized that would make the codes too long. I talked a bit about efficiency, and then they did the summary problems (now HO #9).

If there is time, the following can be used.

The main drawback of this system is that the length of a message is multiplied by 3. So, a file which would take 5 minutes to download without the error correction would now take 15 minutes. Is there a better way?

The criterion for the letter-to-three-letter-word code and the one-bit-to-three-bits codes to be error-correcting is that they have at least three 'bits' different. It turns out we can do better if we encode not single bits at a time, but larger groups of bits. On TSP #27 you can guide them successively through attempts to encode the 4 possible 2-bit stirngs with 3, 4 or 5 bits. On Handout #8, as shown on TSP #28, they are asked to find four strings of 5 bits so that any *two* bits can be encoded with 5 bits (which is only a factor of 2.5 instead of 3 in length expansion) and so that each pair of strings differs by *at least* three bits. The 5-cube is shown, and they are encouraged to try to find these strings on the cube. There is a nice graphical interpretation of what it means that they differ on at least three bits, namely that the distance between them on the graph is at least 3. This actually helps to find the code. The answer (for example) is `00000, 11100, 11011, 00111.`

Finally, you can show them the 7-cube on the last TSP #29, encoding all possible 4-bit strings with 7-bits. TSP #30 does a little analysis...go as far as you wish. The end!

November 14, 2008

Did Codes workshop this morning for 26 teachers from the NJ program. The workshop had not been done for about 7 years, so I made many changes to the materials. The workshop seemed to go well – participants were interested and involved.

The workshop seemed designed for teachers at higher grade levels, so I had to make many modifications to bring the material down to the level of the teachers. However, many of those modifications were not reflected in the materials that they received. For example, the text added to the TSPs were not reflected in the handouts or the resource book. Immediately at the end of the workshop, I prepared a five-page supplementary handout for the Resource Book.

1. I'm not sure why we need the Tampa cryptoquote if we're going to do "The Raven."

2. There should have been a "raven" handout, because many people could not see the TSP very clearly from where they were sitting – as a result, counting the number of occurrences was burdensome. The novelty of the WingDings was not worth the obstacles they created.

3. Here is a cryptoquote – TSP #2. If you are given the key, then it is easy to solve. If you are not given the key, then it is difficult to solve. Here is another cryptoquote – let's try to solve it. This other crypoquote will be the raven, but encoded with letters not symbols, and as a handout (#2) for all participants. We will work together on this for a while (including counting letters), and then each group will complete it on its own.

4. The theme of "secret key" should occur frequently on page 1-3top of notes. There should be a summary of the different uses of codes for conveying and concealing information – while doing this material, I several times was confused about which use of codes my comments fell under. There should be mention of Enigma in WWII and of the bombing of Coventry.

5. After the first paragraph on page 3 of the notes, there should be more detail (for some) of how the prime numbers work – is that still in the category of a substitution code?

6. The public key text should be a handout.

7. It was confusing to ask participants first to decode by the chart and then by the 7-4-2-1-0 method. We should do the method first, and then point out that the lexic ordering gives the numbers 1,2,...9,0 – not treating the chart as a decoding tool at all. The 7-4-2-1-0 method should be at the top of HO #4 (now #2). Question #1 should be omitted, since they were asked to determine the role of the check digit in the session. (Each person was asked to add all ten digits – each table was asked to generalize.)

8. ISBN information should be on HO #5 (now #3), and problem #3 should be moved to the summary set of problems.

9. The activity on HO #6 (now #4) worked well, and many realized that coding A as AAA,

etc., works (the handout and the TSP should be reconciled), and were able to decode the message on TSP #24. The binary message at the bottom should be distributed so that they could work on it.

10. That's as far as I got – that is, we showed that there is a simple error-correcting code. I asked why we didn't triple the zip code and many corrected realized that would make the codes too long. I talked a bit about efficiency, and then they did the summary problems (now HO #6).

Encode the message

"I LOVE YOU"

by replacing each letter

by the letter that comes before it.

You get

"H KNUD XNT"

Your friend can decode the message

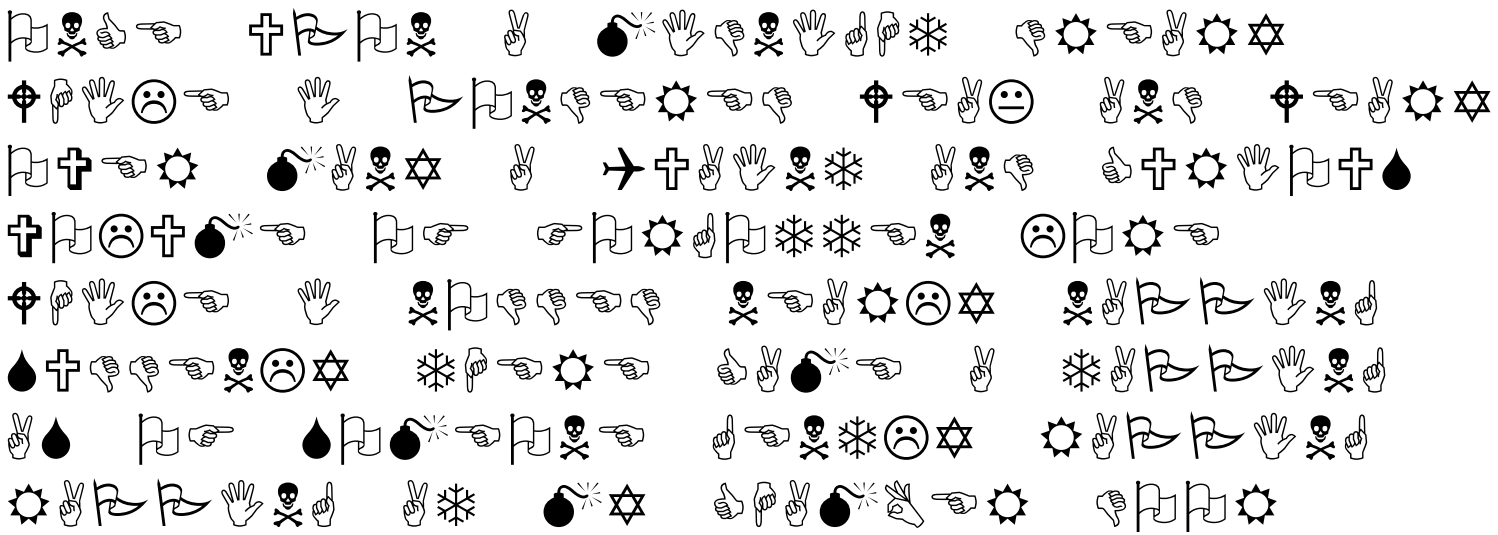
by replacing each letter

by the letter that comes after it.

CRYPTOQUOTE

Tampa Tribune April 7, 1995	***Tampa Tribune*** April 7, 1995
YTDWTXTKB WY LUF YDWFZDF PWLU LUF KAFRLFYL ZSEIFA TG EFLUTNY RZN LUF XFRYL AFYSXLY.-----QTWZDRAF	SOCIOLOGY IS THE SCIENCE WITH THE GREATEST NUMBER OF METHODS AND THE LEAST RESULTS.-----POINCARE

A	R	H	U	O	T	V	M
B	I	I	W	P	Q	W	P
C	D	J	C	Q	J	X	V
D	N	K	H	R	A	Y	B
E	F	L	X	S	Y	Z	O
F	G	M	E	T	L		
G	K	N	Z	U	S		



C r y p t o p o e m

R M G P N W R M B H U K M U C L J K A P B A X,

S L U F P U W R M K P A P K S P B Q B M K S P B A X,

R O P A H B M X B Z N B U M J B M K G N A U R N I

O R F N H P R V V R A C R J J P M F R A P,

S L U F P U M R K K P K, M P B A F X M B W W U M C,

I N K K P M F X J L P A P G B H P B J B W W U M C,

B I R V I R H P R M P C P M J F X A B W W U M C,

A B W W U M C B J H X G L B H D P A K R R A.

How many times does each letter appear in the cryptopoem?

A		H		O		V	
B		I		P		W	
C		J		Q		X	
D		K		R		Y	
E		L		S		Z	
F		M		T			
G		N		U			

Order and Frequency of Single Letters

E 12.31%	L 4.03%	B 1.62%
T 9.59	D 3.65	G 1.61
A 8.05	C 3.20	V 0.93
O 7.94	U 3.10	K 0.52
N 7.19	P 2.29	Q 0.20
I 7.18	F 2.28	X 0.20
S 6.59	M 2.25	J 0.10
R 6.03	W 2.03	Z 0.09
H 5.14	Y 1.88	

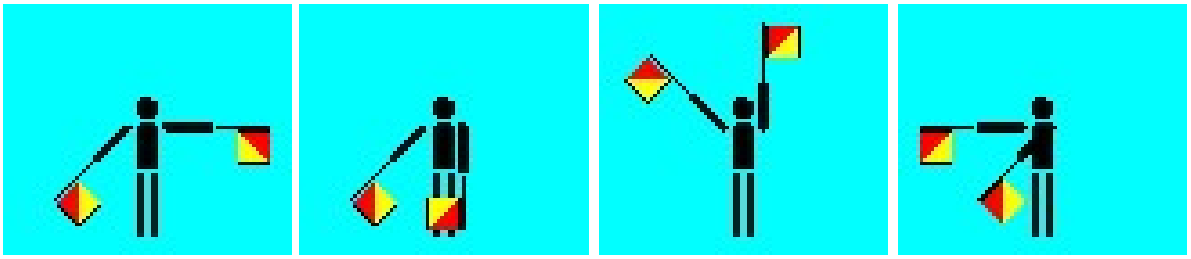
Order and Frequency of Leading DIGRAMS

TH	3.15%	TO	1.11%	SA	0.75%	MA	0.56%
HE	2.51	NT	1.10	HI	0.72	TA	0.56
AN	1.72	ED	1.07	LE	0.72	CE	0.55
IN	1.69	IS	1.06	SO	0.71	IC	0.55
ER	1.54	AR	1.01	AS	0.67	LL	0.55
RE	1.48	OU	0.96	NO	0.65	NA	0.54
ES	1.45	TE	0.94	NE	0.64	RO	0.54
ON	1.45	OF	0.94	EC	0.64	OT	0.53
EA	1.31	IT	0.88	IO	0.63	TT	0.53
TI	1.28	HA	0.84	RT	0.63	VE	0.53
AT	1.24	SE	0.84	CO	0.59	NS	0.51
ST	1.21	ET	0.80	BE	0.58	UR	0.49
EN	1.20	AL	0.77	DI	0.57	ME	0.48
ND	1.18	RI	0.77	LI	0.57	WH	0.48
OR	1.13	NG	0.75	RA	0.57	LY	0.47

Order of Leading TRIGRAMS

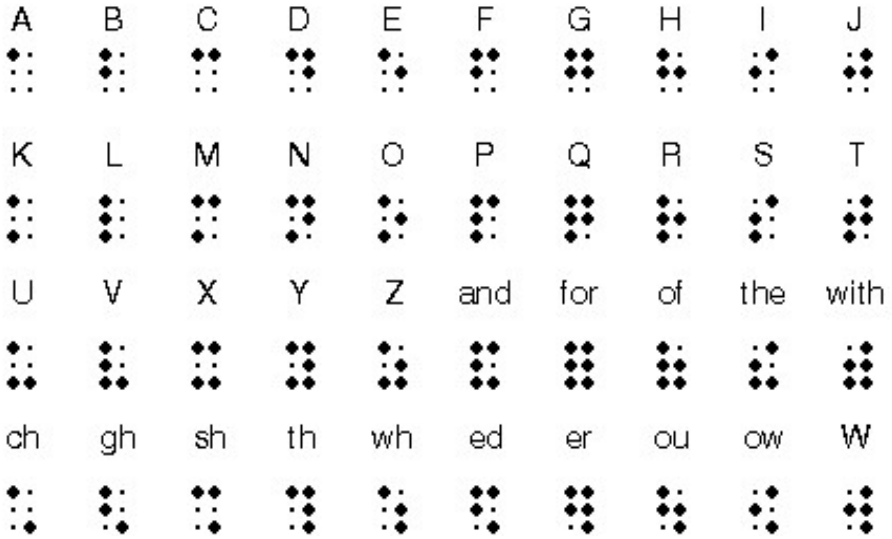
THE AND THA ENT ION TIO FOR NDE HAS NCE EDT TIS OFT STH MEN

Semaphore



M *A* *T* *H*

Braille



Morse Code and Phonetic Alphabets							
Letter	*Morse*	*English*		*Letter*	*English*	*Morse*	
A	*.-*	*Andrew*		*N*	*Nelli*	*--*	
B	*-...*	*Benjamin*		*O*	*Oliver*	*-.*	
C	*-.-.*	*Charlie*		*P*	*Peter*	*.--.*	
D	*-..*	*David*		*Q*	*Queenie*	*--.-*	
E	*.*	*Edward*		*R*	*Robert*	*.-.*	
F	*..-.*	*Frederick*		*S*	*Sugar*	*...*	
G	*--.*	*George*		*T*	*Tommie*	*-*	
H	*....*	*Harry*		*U*	*Uncle*	*..-*	
I	*..*	*Isaac*		*V*	*Victor*	*...-*	
J	*.---*	*Jack*		*W*	*William*	*.--*	
K	*-.-*	*King*		*X*	*Xmas*	*-..-*	
L	*.-..*	*Lucy*		*Y*	*Yellow*	*-.--*	
M	*--*	*Mary*		*Z*	*Zebra*	*--..*	

ZIMMERMAN TELEGRAM – 1917

Sending and receiving messages using a "secret key"

You and I both have the key.

No one else has the key.

You encode the message using the key.

You send the message to me.

I decode the message using the key.

Problems:

1. How do I get the key to you?

2. If I am communicating with many people

...

2a. I have to keep track of lots of keys

2b. I have to know who sent the message

Sending and receiving messages using a "public key"

I have a public key and a secret key.

I tell the world my public key.

No one else knows my secret key.

Everyone encodes their messages to me using the same public key.

I decode all messages using my secret key.

Example

My public key might be a 200-digit number M that is the product of two 100-digit prime numbers A and B.

Your credit card # is encoded using M.

You send the message "charge $100 to my account."

Only someone who knows A and B can decode the message.

Someone who can factor M into A and B can intercept the message.

That is essentially impossible – no one knows how to factor large numbers efficiently.

Postal Codes.

Each ZIP+4 code is converted into a sequence of bars – some long bars and some short bars. There are actually 52 bars, beginning and ending with a long bar, and including ten groups of 5 bars in the middle.

Why are there groups of five bars?

Let's see. How many different ways are there to arrange two long and three short vertical line segments in a row?

Here are two of them:

The reason they are in groups of 5 is that there are 10 ways to arrange 2 longs and 3 shorts, and 10 is also the number of digits we need to code! Here is the code:

1 

2 

3 

4 

5 

6 

7 

8 

9 

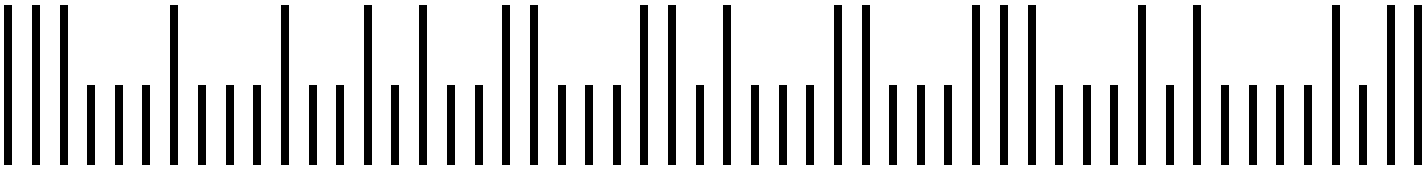
0 

As you can see from the chart, this code is really just a *systematic listing* of the ways to put three short and two long segments together. Which, of course, is just "5 choose 2".

Here's a shortcut – instead of looking up the group of 5 in the table above, just remember the numbers 7-4-2-1-0. To determine the value of a group of 5 bars, just add the two numbers corresponding to the long bars. That is,
a long bar at the left would be treated as 7
a long bar in the next position would be treated as 4
a long bar in the middle would be treated as 2
a long bar in the next position would be treated as 1
and a long bar at the right would be treated as 0.

Thus ❙❙❙❙❙ would be 4 + 1, or 5, and that's the number that's in the chart. When you get 7 + 4, or 11, count that as 0, so that ❙❙❙❙❙ would be 0.

Applying this shortcut to the code below



we get:

“The check digit”

The last group of five bars represents a “check digit” that will detect if the code has been read incorrectly.

When the optical scanner doesn’t work, and the clerk puts it through again and it does work, what has happened is, the first time the scanner misread the bar code, and told the clerk “something’s wrong with this bar code,” and the second time the scanner read the bar code correctly.

How did the scanner know that something was wrong?

The check digit didn’t match the other information!

Can you figure out how the postal check digit works?

The postal check digit is the number which when added to the nine numbers in ZIP+4 gives a multiple of 10.

a)

b)

c)

Error Detection and Correction

When there is an error in a postal bar code, that will be detected if

a. a long bar is replaced by a short bar, or the reverse

b. a number is replaced by a different number.

If the error is of type a, then the error can be corrected – the scanner can figure out what bar has to be fixed.

If the error is of type b, then the error can't be corrected – someone has to look at the letter and make the bar code manually.

ISBN Numbers

Here are some ISBN numbers. What patterns do you see?

0-13-298233-1 (Excursions in Modern Mathematics)
0-394-86580-4 (The Butter Battle Book)
0-671-74468-2 (Star Trek: Legacy)
0-471-02865-7 (Graph Coloring Problems)
0-395-68996-1 (The Love Letter)
0-394-80016-8 (Green Eggs and Ham)
0-7167-1186-9 (The Fractal Geometry of Nature)

The first group of digits gives the language in which the book was written.

The second group of digits is the code of the publisher.

The third group of digits is the publisher's number of the book.

The last digit is a "check digit."

For example, the publisher's code for "Excursions" is 13 and this book is the 298,233rd book they published.

How does the check digit work for ISBN numbers?

Write the ten digits in the following slots.

___	___	___	___	___	___	___	___	___	___
10	9	8	7	6	5	4	3	2	1

Multiply the number beneath the slot, and add.

The check digit is selected so that the total is a multiple of 11.

For example, with the ISBN number
0-7167-1186-9, you get

_0_	_7_	_1_	_6_	_7_	_1_	_1_	_8_	_6_	_9_
10	9	8	7	6	5	4	3	2	1

The sum is

$0 + 63 + 8 + 42 + 42 + 5 + 4 + 24 + 12 + 9 = 209$

which is a multiple of 11; $209 = 11 \times 19$.

This will detect a single error – suppose for example you incorrectly wrote down
0-7168-1186-9 (typing 7168 instead of 7167)

you would get

_0_	_7_	_1_	_6_	_8_	_1_	_1_	_8_	_6_	_9_
10	9	8	7	6	5	4	3	2	1

The sum is
0 + 63 + 8 + 42 + 48 + 5 + 4 +24 +12 + 9 = 215
which is not a multiple of 11; 215 = (11 x 19) + 6

This will also catch a transposition error (which is not caught by ZIP code scanners). For example, if you wrote 7176 instead of 7167, then you would get

_0_	_7_	_1_	_7_	_6_	_1_	_1_	_8_	_6_	_9_
10	9	8	7	6	5	4	3	2	1

The sum is
0 + 63 + 8 + 49 + 36 + 5 + 4 +24 +12 + 9 = 210, which is not a multiple of 11.

What possible one-digit changes could you make to 0-7168-1186-9 and get a correct ISBN number?

Adding 6 to first digit adds 60 to 215 to get 275

Adding 2 to third digit adds 16 to 215 to get 231

Subtracting 4 from fourth digit subtracts 28 to get 187

Subtracting 1 from fifth digit subtracts 6 to get 209

Adding 1 to sixth digit adds 5 to get 220

Subtracting 2 from eighth digit subtracts 6 to get 209

Subtracting 3 from ninth digit subtracts 6 to get 209

Subtracting 6 from tenth digit subtracts 6 to get 209

Since there are a lot of possible errors that could give this "ISBN," ISBN cannot "correct" the "error."

ISBN is not an error-correcting code.

However ...
... if you assume that the error is the result of transposing two adjacent numbers rather than typing one number incorrectly, then the error can frequently be corrected.

For example, we just considered the question,

"What possible one-digit changes could you make to 0-7168-1186-9 and get a correct ISBN number?"

And found that there were eight possible answers.

If instead we considered the question,

What possible transpositions could you make to 0-7168-1186-9 and get a correct ISBN number?

there is only one possible answer.

So even though ISBN is not an error-correcting code, it often corrects transposition errors very well.

AN ERROR CORRECTING CODE?

A	ARK
B	BAD
C	CAT
D	DOT
E	EEL
F	FAT
G	GOT
H	HIP
I	INK
J	JET
K	KID
L	LIP
M	MOP

N	NOT
O	OAK
P	PAN
Q	QED
R	RUT
S	SAT
T	TIC
U	USE
V	VEX
W	WAX
X	XED
Y	YET
Z	ZIP

In order to try to correct a single-letter error in transmitting a very important message, Alice decides to substitute a three-letter word for each letter and send that. Thus "HELP" becomes "HIP EEL LIP PAN". Can this encoded message always be decoded correctly despite an error in which exactly one letter is changed?

Conclusion: If each letter is replaced by three letters, then any two three-letter codes must differ on all three letters.

So if we replace A by AAA, B by BBB, C by CCC, etc. we would have an error-correcting code – that is, you would be able to correct any single error.

For example, you would be able to figure out:

SHHEEPLLIKPP

Note that you are able to detect and correct any single error in each group of three letters.

Messages from space are sent using 0 and 1. Here the code used is:

0➞000 1➞111

Thus `11001` becomes `111111000000111`

Now, suppose a message encoded as above is sent from a satellite and received as below. Can you decode it to recover the intended string of bits?

```
111000111001000000111000000111101110111
111000000000100001000000100000001000111
```

There are 4 two-bit strings:

A	00
B	01
C	10
D	11

Let us try to replace these strings with other, longer code strings of 0s and 1s so that every pair of code strings differ on at least three bits.

Can you do this with three-bit strings? No!

Can you do this with four-bit strings? Again no, but why?

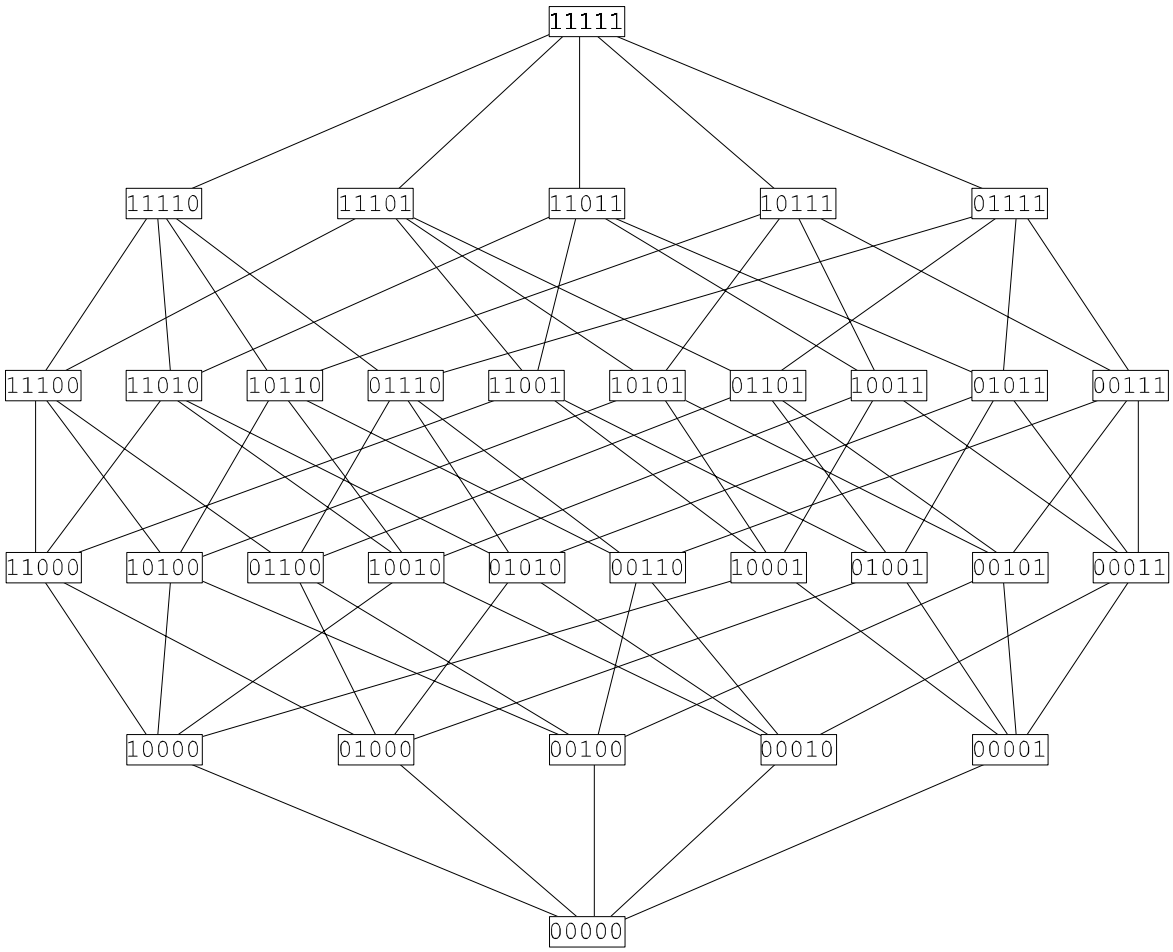
Can you do this with five-bit strings?

Let's try!

(Note, calling a string a 'two-bit string' is not an insult!)

How many five-bit 0-1 strings are there?

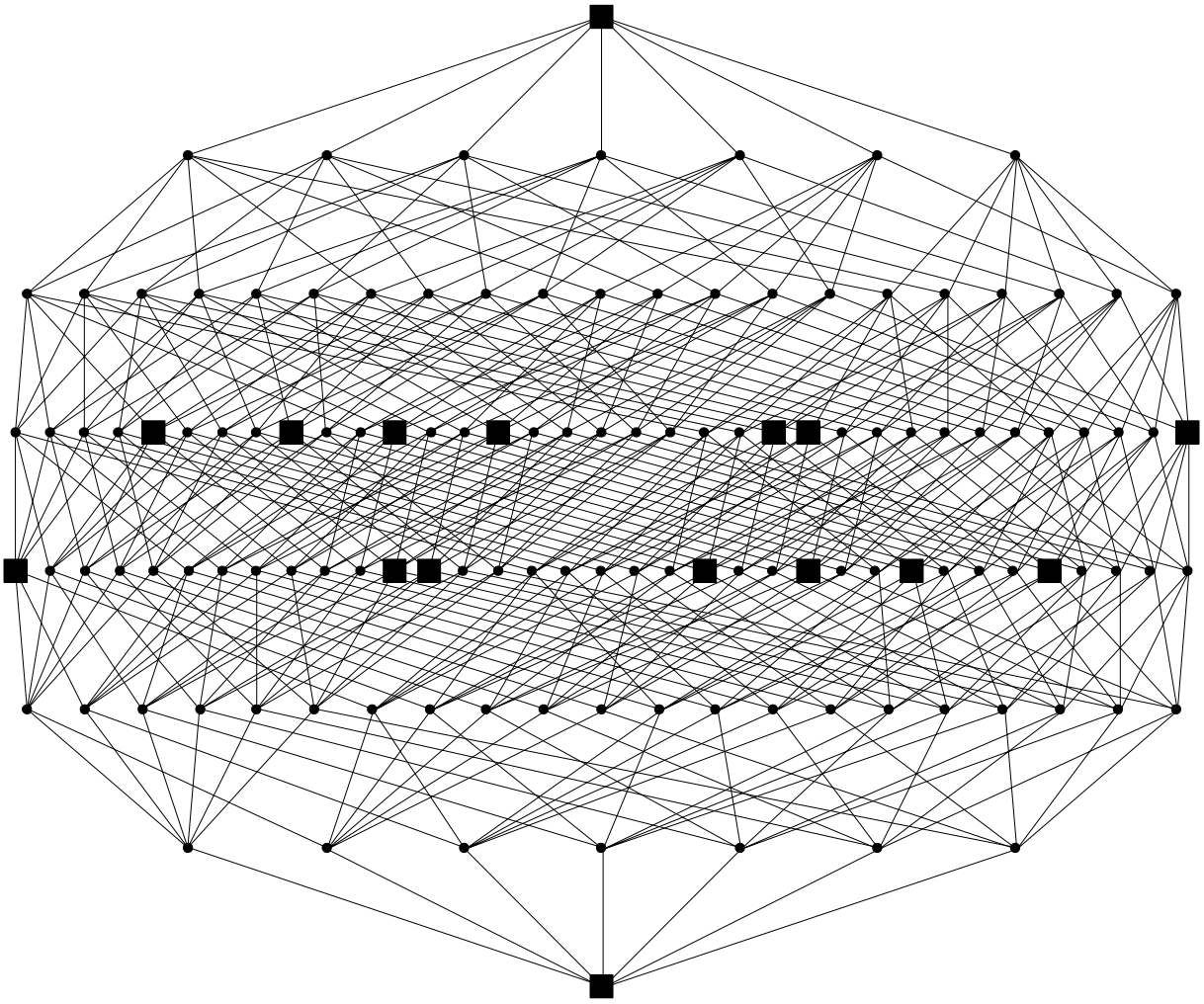
Here they are:



Can you see the rule for connecting two vertices in this graph? Can you find the postal code? What does finding an error-correcting code correspond to on this graph, if we require that any two code words differ on at least three bits?

What if we wished to find a way to encode all possible 4-bit strings?

It turns out we can do it with seven bits, as shown below:



Encoding 4 bits with 7
Correcting a 1 bit error per block.

0000 A	0000000	1000 I	1101010
0001 B	1110000	1001 J	1100101
0010 C	1001100	1010 K	1010110
0011 D	1000011	1011 L	1011001
0100 E	0100110	1100 M	0111100
0101 F	0101001	1101 N	0110011
0110 G	0011010	1110 O	0001111
0111 H	0010101	1111 P	1111111

Any pair of two code words differ on at least 3 bits.

Thus this code can correct a one bit error per block of 7.

This is called the “Hamming Code” on the 7-cube.

It is quite efficient because the multiplication factor is only 7/4, or 1.75.

Handout #1 – Substitution Codes

1. A student assigned the numbers 1 through 26 to the letters A-Z in increasing order. Can you decode the following message?:

   5 1 19 25 3 15 4 5.

2. The following message was encoded by replacing each letter with the letter that comes before it in the alphabet. What does the message say?

   BNTKC XNTQ RSTCDMSR CDBNCD SGHR? VGZS ZANTS "Z"?

3. Another student assigned numbers to the letters as follows: A--1, B--4, C--7, D--10, etc... Continue the pattern and decode the following message:

   13 61 34 13 52 7 25 52 7 61 25 58?

Handout #2 – Cryptopoem

Can you decode this cryptopoem?

R M G P N W R M B H U K M U C L J K A P B A X,

S L U F P U W R M K P A P K S P B Q B M K S P B A X,

R O P A H B M X B Z N B U M J B M K G N A U R N I

O R F N H P R V V R A C R J J P M F R A P,

S L U F P U M R K K P K, M P B A F X M B W W U M C,

I N K K P M F X J L P A P G B H P B J B W W U M C,

B I R V I R H P R M P C P M J F X A B W W U M C,

A B W W U M C B J H X G L B H D P A K R R A.

How many times does each letter appear in the cryptopoem?

A		H		O		V	
B		I		P		W	
C		J		Q		X	
D		K		R		Y	
E		L		S		Z	
F		M		T			
G		N		U			

Handout #3 – Secret keys and public keys

Sending and receiving messages using a "secret key"

You and I both have the key.

No one else has the key.

You encode the message using the key.

You send the message to me.

I decode the message using the key.

Problems:

1. How do I get the key to you?

2. If I am communicating with many people ...

2a. I have to keep track of lots of keys

2b. I have to know who sent the message

Sending and receiving messages using a "public key"

I have a public key and a secret key.

I tell the world my public key.

No one else knows my secret key.

Everyone encodes their messages to me using the same public key.

I decode all messages using my secret key.

Example

My public key might be a 200-digit number M that is the product of two 100-digit prime numbers A and B. Your credit card # is encoded using M.

You send the message "charge $100 to my account."

Only someone who knows A and B can decode the message.

Someone who can factor M into A and B can intercept the message.

That is essentially impossible – no one knows how to factor large numbers efficiently.

Handout #4 – Postal Codes

1. Here are several ZIP+4 Post codes without the check digits. What should the check digit be in each case?

a) 08854-3113-? b) 40004-3225-? c) 14347-9133-? d) 91225-3105-?

2. Decode each of the following Postal Zip+4 codes.

a) 

b) 

c) 

3. a. Find the error in b above and correct it.

b. Why can't you find the error in c above?

Handout #5 – ISBN Codes

1. Which of the following is the real ISBN code for "Graph Theory" by Béla Bollabás?

a) `3-540-93099-2` b) `3-540-90399-2`

2. The ISBN number `0-7168-1186-9` is incorrect. Assuming that the only error was that a single digit had been incorrectly entered, what was the correct, original ISBN number?

Handout #6 – Error-Correcting Codes

AN ERROR CORRECTING CODE?

A	ARK
B	BAD
C	CAT
D	DOT
E	EEL
F	FIG
G	GOT
H	HIP
I	INK
J	JET
K	KID
L	LIP
M	MOP

N	NOT
O	OAK
P	PAN
Q	QED
R	RUT
S	SAT
T	TIC
U	USE
V	VEX
W	WAX
X	XED
Y	YET
Z	ZIP

In order to try to correct a single-letter error in transmitting a very important message, Alice decides to substitute a three-letter word for each letter and send that. Thus "HELP" becomes "HIP EEL LIP PAN". Can this encoded message always be decoded correctly despite an error in which exactly one letter is changed?

Can you design a set of three-letter words which will give the desired error-correcting code? You can stretch our dictionary a bit and use any three-letter combo that you wish as a "word".

Why can't this be done with sets of *two*-letter words, even if you allow such combinations as "pl" and "xx" and "zi" to be considered words?

Handout #7 – Error-Correcting Codes

Messages from space are sent using 0 and 1. Here the code used is:

0➞000 1➞111

Thus `11001` becomes `111111000000111`

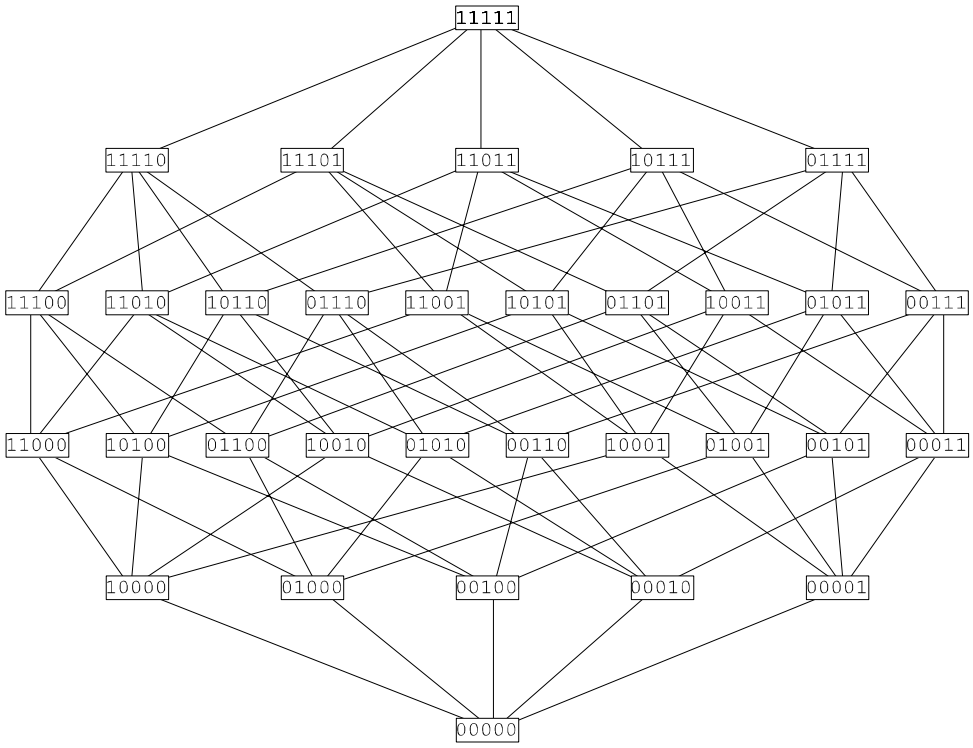
Now, suppose a message encoded as above is sent from a satellite and received as below. Decode it to recover the intended string of 0s and 1s.

```
111000111001000000111000000111101110111
111000000000100001000000100000001000111
```

Handout #8 – Error-Correcting Codes

Here we will try to find five-bit code strings to replace the two-bit strings A, B, C and D, in such a way that each pair of code strings differ on at least three bits.

We might as well assume that one of the code strings is 00000. Do you see what the figure above represents? What information does this graph (called the 5-dimensional cube) represent?



A) Circle the other strings that choosing 00000 eliminates as possible code-words.
B) Can you find 3 other code strings which all differ from 00000 and from each other by at least three bits?

Handout #9 – Summary

1. The following message was encoded using a method similar to that used in problem #3 on Handout #1. Can you decode the message?

   35 47 35 43 19 79 59 23 35 55 15 31 3 51 35 47 79 59 55 63 3 79 31 75!

2. Can you figure out the following message using a substitution code without knowing in advance how to decode it?

W urmf r nafre lurl eb gtsa xwllxf duwxnafz pwxx t

zf

nrb xwmf wz r zrlwtz pufaf lufb pwxx ztl if csnkfn i

b

luf dtxta tg lufwa yhwz isl ib luf dtzlfzl tg lufwa

durardlfa.

3. a. The ISBN of "Green Eggs and Ham" is 0-394-80016-8. Verify that this is a correct ISBN number by making the usual calculation:

___	___	___	___	___	___	___	___	___	___
10	9		8	7	6	5	4	3	2	1

   b. What would be the effect on the calculation if the 9 and 4 were transposed?
   c. What would be the effect on the calculation if the 4 and 8 were transposed?
   d. What would be the effect on the calculation if the 1 and 6 were transposed?
   e. What pattern do you find in your answers to b-d?
   f. Explain why this effect takes place on a correct ISBN if two adjacent numbers L and R are transposed? (The pattern is that if you reverse two adjacent numbers LR in the ISBN, so that you mistakenly get RL, then the following happens – if R is larger, then R-L is added to the sum, whereas if L is larger, L-R is subtracted from the sum.)

g. Can you use that information to figure out what is the correct ISBN for "The Love Letter" if the number you have is 0-359-68996-1 and you know that two adjacent numbers have been transposed?

4. Can you design a check-digit encoding system that will detect an error in a birthdate entered in the format mm-dd-yy-c, where "mm" is the month (01=Jan, 02=Feb, ..., 12=Dec), "dd" is the date (01, 02, .., or 31), "yy" is the year and "c" is the check digit? The system should be able to detect an error when exactly one of the digits has been entered incorrectly.

I have a dream that my four little children will one day live in a nation where they will not be judged by the color of their skin but by the content of their character.

CODES

Resource Materials

Follow-up session

November/December 2008

CRYPTOQUOTE

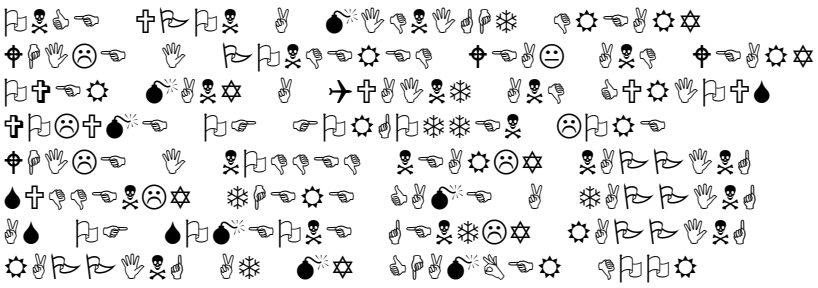
Tampa Tribune April 7, 1995

YTDWTXTKB WY LUF YDWFZDF
PWLU LUF KAFRLFYL ZSEIFA
TG EFLUTNY RZN LUF XFRYL
AFYSXLY.-----QTWZDRAF

Tampa Tribune April 7, 1995

SOCIOLOGY IS THE SCIENCE
WITH THE GREATEST NUMBER
OF METHODS AND THE LEAST
RESULTS.-----POINCARE

A	R V	H M	U	O	T
B	I W	I P	W	P	Q
C	D X	J V	C	Q	J
D	N Y	K B	H	R	A
E	F Z	L O	X	S	Y
F	G	M	E	T	L
G	K	N	Z	U	S



The solution to the code above (in Word Perfect Wingding font) is given below. But for those who would like to opportunity to decode it for themselves, the answer is given in a shift substitution code. Each letter is shifted three letters later in the alphabet.

Wkh udyhq, eb H. D. Srh.

Order and Frequency of Leading DIGRAMS

TH	3.15%	TO	1.11%	SA	0.75%	MA	0.56%
HE	2.51	NT	1.10	HI	0.72	TA	0.56
AN	1.72	ED	1.07	LE	0.72	CE	0.55
IN	1.69	IS	1.06	SO	0.71	IC	0.55
ER	1.54	AR	1.01	AS	0.67	LL	0.55
RE	1.48	OU	0.96	NO	0.65	NA	0.54
ES	1.45	TE	0.94	NE	0.64	RO	0.54
ON	1.45	OF	0.94	EC	0.64	OT	0.53
EA	1.31	IT	0.88	IO	0.63	TT	0.53
TI	1.28	HA	0.84	RT	0.63	VE	0.53
AT	1.24	SE	0.84	CO	0.59	NS	0.51
ST	1.21	ET	0.80	BE	0.58	UR	0.49
EN	1.20	AL	0.77	DI	0.57	ME	0.48
ND	1.18	RI	0.77	LI	0.57	WH	0.48
OR	1.13	NG	0.75	RA	0.57	LY	0.47

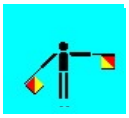
Order and Frequency of Single Letters

E 12.31%	**L 4.03%**	**B 1.62%**
T 9.59	**D 3.65**	**G 1.61**
A 8.05	**C 3.20**	**V 0.93**
O 7.94	**U 3.10**	**K 0.52**
N 7.19	**P 2.29**	**Q 0.20**
I 7.18	**F 2.28**	**X 0.20**
S 6.59	**M 2.25**	**J 0.10**
R 6.03	**W 2.03**	**Z 0.09**
H 5.14	**Y 1.88**	

Order of Leading TRIGRAMS

THE AND THA ENT ION
TIO FOR NDE HAS NCE
EDT TIS OFT STH MEN

Semaphore



M A T H

Braille

To determine the value of a configuration,
use the place values: 7 4 2 1 0
With the convention that “7+4=0".

Morse Code and Phonetic Alphabets

Letter	*Morse*	*English*		*International*		*NATO & international aviation (police)*	*American*				
							Italian	*German*			
A	.-	*Andrew*	*Amsterdam*		*Alfa*			*Adam*	*Ancona*	*Anton*	
Ä	.-.-									*Ärger*	
Á	.--.-										
Å	.--.-										
B	-...	*Benjamin*		*Baltimore*				*Bravo*	*Boy*	*Bolog na*	*Berta*
C	-.-.	*Charlie*	*Casablanca*		*Charlie*			*Charlie*	*Como*	*Cäsar*	
Ch	----									*Charlotte*	
D	-..	*David*	*Denmark*		*Delta*			*David*	*Domodossola*	*Dora*	
E	.	*Edward*	*Edison*	*Echo*				*Edward*	*Empoli*	*Emil*	
É	..-..										
F	..-.	*Frederick*		*Florida*		*Foxtrot*	*Frank*	*Firenze*	*Friedrich*		
G	--.	*George*	*Gallipoli*			*Golf*	*George*	*Genova*	*Gustav*		
H	....	*Harry*	*Havana*	*Hotel*		*Henry*	*Hotel*	*Heinrich*			
I	..	*Isaac*	*Italy*	*India*		*Ida*	*Imola*	*Ida*			
J	.---	*Jack*	*Jerusalem*			*Juliet*	*John*	*I lunga*	*Julius*		
K	-.-	*King*	*Kilogram*			*Kilo*	*King*	*Kursaal*	*Kaufmann*		
L	.-..	*Lucy*	*Liverpool*			*Lima*	*Lincoln*	*Livorno*	*Ludwig*		
M	--	*Mary*	*Madagascar*			*Mike*	*Mary*	*Milano*	*Martha*		
N	-.	*Nelli*	*New York*			*November*		*Nora*	*Napoli*	*Nordpol*	
Ñ	--.--										
O	---	*Oliver*	*Oslo*	*Oscar*		*Ocean*	*Otranto*	*Otto*			
Ö	---.							*Ökkonom*			
P	.--.	*Peter*	*Paris*	*Papa*		*Paul*	*Padova*	*Paula*			
Q	--.-	*Queenie*	*Quebec*	*Quebec*		*Queen*	*Quarto*	*Quelle*			
R	.-.	*Robert*	*Roma*	*Romeo*		*Robert*	*Roma*	*Richard*			
S	...	*Sugar*	*Santiago*			*Sierra*	*Sam*	*Savona*	*Samuel*		
Sch								*Schule*			
T	-	*Tommy*	*Tripoli*	*Tango*		*Tom*	*Torino*	*Theodor*			
U	..-	*Uncle*	*Uppsala*			*Uniform*		*Union*	*Udine*	*Ulrich*	
Ü	..--							*Übermut*			

V	*...-*	*Victor*	*Valencia*		*Victor*	*Victor*	*Venezia*	*Viktor*
W	*.--*	*William*	*Washington*		*Whisky*	*William*	*Washington*	*Wilhelm*
X	*-..-*	*Xmas*	*Xanthippe*		*X-ray*	*X-ray*	*Ics*	*Xanthippe*
Y	*-.--*	*Yellow*	*Yokohama*		*Yankee*	*Young*	*York*	*Ypsilon*
Z	*--..*	*Zebra*	*Zürich*	*Zulu*	*Zebra*	*Zara*	*Zeppelin*	

1 
2 
3 
4 
5 
6 
7 
8 
9 
0 

As you can see from the chart, this code is really just a *systematic listing* of the ways to put three short and two long segments together. Which, of course, is just "5 choose 2".

0-7165-1186-9

0	7	1	6	5	1	1	8	6	9
10	9	8	7	6	5	4	3	2	1

The sum is 0+63+8+42+30+5+4+24+12+9 = 197

And 197/11 = 17 remainder 10

There are many errors that could have caused this, and many "corrections" that would give no remainder:

- make the 9 an "x" in the "1" position
- add 3 in the "4" position
- add 2 in the "6" position
- subtract 7 in the "3" position
- etc...

Thus ISBN numbers can not "correct" an error.

ISBN Numbers

Here are some ISBN numbers. What patterns do you see?

`0-13-298233-1` (Excursions in Modern Mathematics)
`0-394-86580-4` (The Butter Battle Book)
`0-671-74468-2` (Star Trek: Legacy)
`0-471-02865-7` (Graph Coloring Problems)
`0-395-68996-1` (The Love Letter)
`0-394-80016-8` (Green Eggs and Ham)
`0-7167-1186-9` (The Fractal Geometry of Nature)

Again, the last digit is an error-detection digit.
Write the digits in the slots,

___	___	___	___	___	___	___	___	___	___
10	9	8	7	6	5	4	3	2	1

Multiply by the number beneath the slot, and add. The result should be a multiple of 11.

This will detect a single error as well as a "transposition," which is a common error when people enter ISBN numbers.

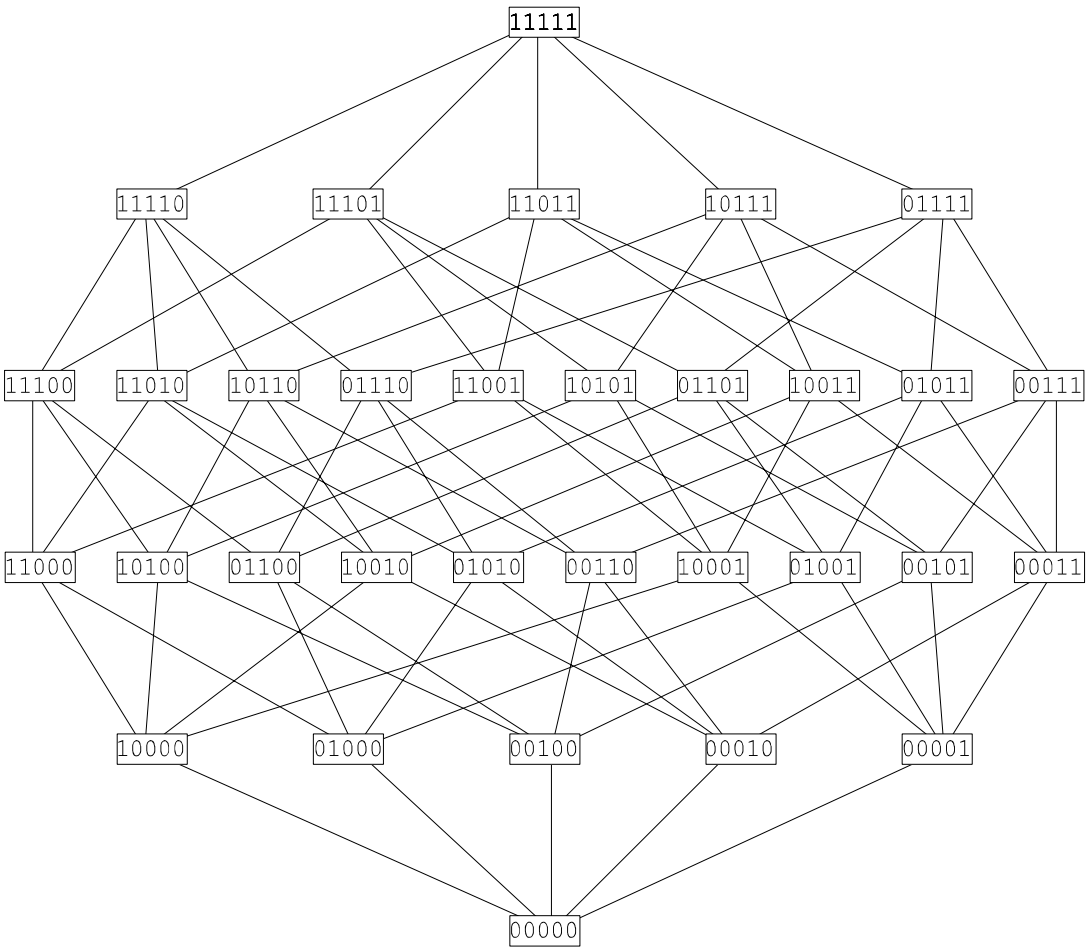
Can you "correct" this postal bar code?

This is exercise 2. b from Handout #4.

How many five-bit 0-1 strings are there?

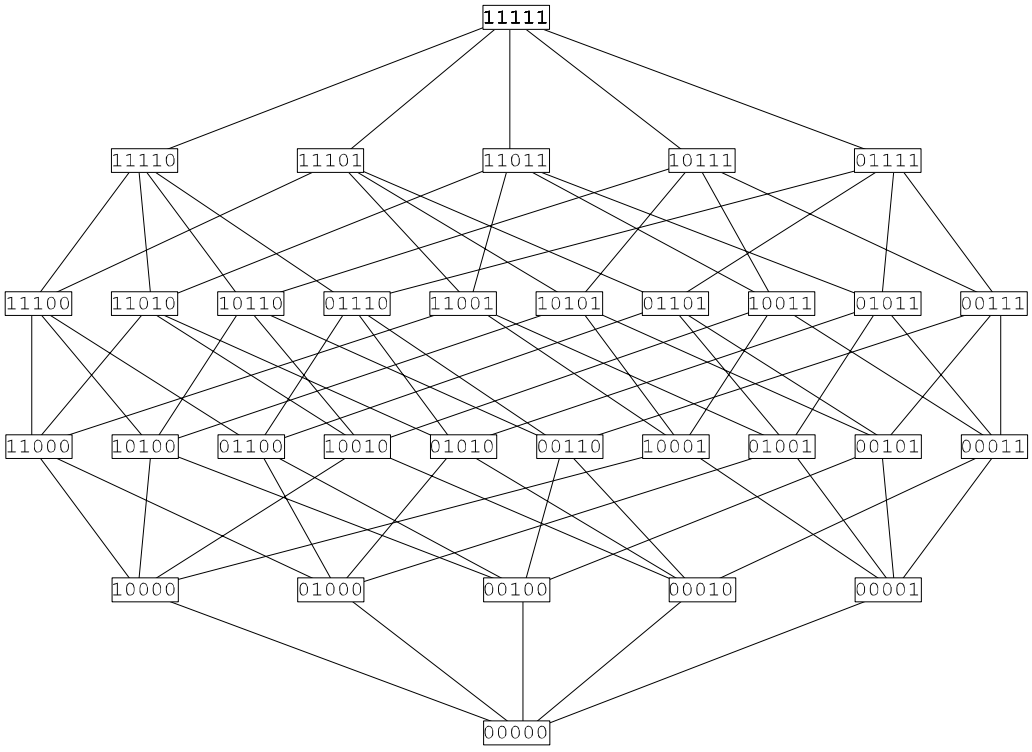
Here they are:

Can you see the rule for connecting two vertices in this graph? Can you find the postal code? Can you find 4 “code strings” among the 32 in the figure, so that any pair of them differs on at least three bits.



AN ERROR CORRECTING CODE?

A	ARK	N	NOT
B	BAD	O	OAK
C	CAT	P	PAN
D	DOT	Q	QED
E	EEL	R	RUT
F	FIG	S	SAT
G	GOT	T	TIC
H	HIP	U	USE
I	INK	V	VEX
J	JET	W	WAX
K	KID	X	XED
L	LIP	Y	YET
M	MOP	Z	ZIP



The following pages contain the text of a number of transparencies in the workshop.

A.

Encode the message "I LOVE YOU" by replacing each letter by the letter that comes before it.

You get "H KNUD XNT" Your friend can decode the message by replacing each letter by the letter that comes after it.

If your friend doesn't know the key (the coding system), then he might think that the message reads "I LOVE ROB."

B. Zimmerman Telegram.

The Zimmerman telegram was sent by the German Foreign Secretary, Arthur Zimmermann, on January 16, 1917, to the German ambassador in the United States of America, Johann von Bernstorff, at the height of World War I. On January 19, Bernstorff, per Zimmermann's request, forwarded the Telegram to the German ambassador in Mexico, Heinrich von Eckardt. The Telegram instructed Ambassador Eckardt that if the United States appeared likely to enter the war he was to approach the Mexican government with a proposal for military alliance. He was to offer to help Mexico reclaim Texas, New Mexico, and Arizona that it had lost in the Mexican-American War. The Zimmermann Telegram was intercepted and decoded by the British cryptographers. The revelation of its contents in the American press on March 1 caused public outrage that contributed to the United States' declaration of war against Germany on April 6.

C. Sending and receiving messages using a "secret key"

You and I both have the key.
No one else has the key.
You encode the message using the key.
You send the message to me.
I decode the message using the key.

Problems:
1. How do I get the key to you?
2. If I am communicating with many people ...
2a. I have to keep track of lots of keys
2b. I have to know who sent the message

D. Sending and receiving messages using a "public key"

I have a public key and a secret key.
I tell the world my public key.
No one else knows my secret key.
Everyone encodes their messages to me using the same public key.
I decode all messages using my secret key.

Example
My public key might be a 200-digit number M that is the product of two 100-digit prime numbers A and B.
Your credit card # is encoded using M.
You send the message "charge $100 to my account."
Only someone who knows A and B can decode the message.
Someone who can factor M into A and B can intercept the message.
That is essentially impossible – no one knows how to factor large numbers efficiently.

E. Postal Codes.

Each ZIP+4 code is converted into a sequence of bars – some long bars and some short bars. There are actually 52 bars, beginning and ending with a long bar, and including ten groups of 5 bars in the middle. Moreover, each group has two longs and three shorts.

Why are there groups of five bars? Let's see. How many different ways are there to arrange two long and three short vertical line segments in a row? Here are two of them: or

There are altogether ten such arrangements, since any arrangement involves choosing two of the five slots in which we will use longs, and there are "5 choose 2," or 5x4/2, or 10 ways of doing that.

So now we see why we use groups of five bars: We need 10 codes, one for each of the digits between 0 and 9, and there are 10 ways to make a group out of two longs and three shorts, so we can use the ten groups of five, with two long and three short, as codes for our ten digits.

Which code goes with each digit. Here's a shortcut – just remember the numbers 7-4-2-1-0. To determine the value of a group of 5 bars, just add the two numbers corresponding to the long bars. Thus  would be 4 + 1, or 5, and that's the number that's in the chart. When you get 7 + 4, or 11, count that as 0, so that would be 0.

F. The check digit (for Postal Codes).

The last group of five bars represents a "check digit" that will detect if the code has been read incorrectly. When the optical scanner doesn't work, and the clerk puts it through again and it does work, what has happened is, the first time the scanner misread the bar code, and told the clerk "something's wrong with this bar code," and the second time the scanner read the bar code correctly.

How did the scanner know that something was wrong? The check digit didn't match the other information!

Can you figure out how the postal check digit works? It's very simple. The number in the tenth place is chosen so that the sum of the ten digits is a multiple of 10.

G. Error Detection and Correction (for Postal Codes)

When there is an error in a postal bar code, that will be detected if

a. a long bar is replaced by a short bar, or the reverse
b. a number is replaced by a different number.

If the error is of type a, then the error can be corrected – the scanner can figure out what bar has to be fixed. If

the error is of type b, then the error can't be corrected – someone has to look at the letter and make the bar code manually.

H. ISBN numbers

A typical ISBN number looks like 0-7167-1186-9 . The first group of digits gives the language in which the book was written. The second group of digits is the code of the publisher. The third group of digits is the publisher's number of the book. The last digit is a "check digit."

How does the check digit work for ISBN numbers?

Write the ten digits in the following slots.

___	___	___	___	___	___	___	___	___	___
10	9	8	7	6	5	4	3	2	1

Multiply the number beneath the slot, and add.

The check digit is selected so that the total is a multiple of 11.

For example, with the ISBN number 0-7167-1186-9, you get

_0_	_7_	_1_	_6_	_7_	_1_	_1_	_8_	_6_	_9_
10	9	8	7	6	5	4	3	2	1

The sum is 0 + 63 + 8 + 42 + 42 + 5 + 4 +24 +12 + 9 = 209, which is a multiple, 19x11 of 11.

This will detect a single error – suppose for example you incorrectly wrote down
0-7168-1186-9 (typing 7168 instead of 7167) you would get

_0_	_7_	_1_	_6_	_8_	_1_	_1_	_8_	_6_	_9_
10	9	8	7	6	5	4	3	2	1

The sum is 0 + 63 + 8 + 42 + 48 + 5 + 4 +24 +12 + 9 = 215
which is not a multiple of 11; 215 = (11 x 19) + 6

This will also catch a transposition error (which is not caught by ZIP code scanners). For example, if you wrote 7176 instead of 7167, then you would get

_0_	_7_	_1_	_7_	_6_	_1_	_1_	_8_	_6_	_9_
10	9	8	7	6	5	4	3	2	1

The sum is 0 + 63 + 8 + 49 + 36 + 5 + 4 +24 +12 + 9 = 210, which is not a multiple of 11.

Moreover, if you assume that the error is the result of transposing two adjacent numbers rather than typing one number incorrectly, then the error can frequently be corrected.

For example, if we consider the question, "What possible one-digit changes could you make to 0-7168-1186-9 and get a correct ISBN number?" – it turns out that there are eight possible answers, so the error can't be corrected.

If, on the other hand, we considered the question, What possible transpositions could you make to 0-7168-1186-9 and get a correct ISBN number? – it turns out that there is only one possible answer.

So even though ISBN is not an error-correcting code, it often corrects transposition errors very well. It turns out that if we accidently transpose two numbers L (left) and R (right) so that instead of LR we type RL, then the sum that we calculated would be R-L greater than the correct sum if R > L and L-R less than the correct sum if L > R. So a transposition of adjacent numbers that differ by 1 would result in a number which is 1 more than (or 1 less than) a multiple of 11. The only occasion where two adjacent numbers differ by 1 in 0-7176-1186-9 is the 7-6 – and if we replace them by 6-7, we get an actual ISBN number. Since the errors in coding ISBN numbers are most often the result of a transposition, it is likely that the original number was 0-7167-1186-9.

I. Error Correcting Codes

If we replace A by AAA, B by BBB, C by CCC, etc. we would have an error-correcting code – that is, you would be able to correct any single error.

For example, you would be able to figure out:

SHHEEPLLIKPP

Note that you are able to detect and correct any single error in each group of three letters.

Messages from space are sent using 0 and 1. Here the code used is:

0➞000 1➞111

Thus `11001` becomes `111111000000111`

Now, suppose a message encoded as above is sent from a satellite and received as below. Can you decode it to recover the intended string of bits?

```
111000111001000000111000000111101110111
111000000000100001000000100000001000111
```