CHECK CODES

by

Gavin Brown*

Have you ever cast a second glance at the ten-digit codes now found on most books? A random volume from my bookshelf carries the message

ISBN 0140050930.

This is the International Standard Book Number in which the first three digits tell us the country of origin and publisher (in this case Penguin, UK) and the next six digits are publisher's code for the book (e.g. series, author, further distinguishing data). The final digit is a check to guard against errors in transcription (or transmission) of the information.

We'll see, in a moment, how this works. First let's pause to consider the trade-off involved. It is much quicker and simpler to order or catalogue a book with

ISBN 014009766X

than to write out

"The Complete Father Brown" by G.K. Chesterton, published by Penguin.

On the other hand a transposition to 019004766X (i.e. the digits 4,9 interchanged) is apparently more serious than one to "The Complehe Fatter Brown" by G.K. Chesterton, published by Penguin. The sheer redundancy of the long-winded message protects its integrity. In this case, however, the check digit (here X standing for 'ten') flags the error. The number 019004766X is easily seen to be impossible as an ISBN, so we know there's a mistake and we can ask for re-transmission of the number. Such a process involves some cost but this is more than compensated by the speed and ease of sending (and storing) lots of book information in this way.

I promised to show how the ISBN's work - let's develop a feel for it as we go.

We work modulo 11 so are concerned with only 0, 1, 2, ..., 9,X (remember that X denotes ten as in Roman numerals). When adding two numbers we use the remainder of the answer when divided by 11. Thus $9 + 8 = 6$, $3 + X = 2$ and so on. To check the number 014009766X we write it vertically and fill two more columns as shown:

---

* Gavin Brown is Professor of Pure Mathematics at the University of New South Wales.

0	0	0
1	1	1
4	5	6
0	5	0
0	5	5
9	3	8
7	X	7
6	5	1
6	0	1
X	X	0

We made the top row 0 0 0 then obtained the remaining entries by adding on an upward rightward diagonal one step at a time.

Let's check the first number I mentioned:

0	0	0
1	1	1
4	5	6
0	5	0
0	5	5
5	X	4
0	X	3
9	8	0
3	0	0
0	0	0

The important point is that the final entry is zero. Now let's check the false number 019004766X

0	0	0
1	1	1
9	X	0
0	X	X
0	X	9
4	3	1
7	X	0
6	5	5
6	0	5
X	X	4

The digit 4 flags the error. The check digit X should not accompany the information 019004766. The appropriate check digit would have been 6 and 0190047666 could represent an ISBN.

How much error protection comes from the check digit? It flags any single wrong digit or any transposition of two digits! That's pretty good going for a single check digit, so how does it work?

Let's design a check system for 3 digit information transfer. In view of what we've seen already we expect to work modulo 5 and so our digits will be 0,1,2,3,4. Here again we take the remainder of each answer after division by 5 (i.e. 3 + 4 = 2, 3 + 2 = 0 etc.). Is it difficult to guess that the string of digits 123 will have the check digit 4? The table below shows that

1	1	1
2	3	4
3	1	0
4	0	[0]

with this check digit we do get the desired final entry 0. Lets work out the check digit for 232:

2	2	2
3	0	2
2	2	4
4	1	[0]

The last row we worked "backwards" from the desirable 0 final entry.

Of course we've cheated and not designed anything. We can at least pretend to work from scratch as follows: First we add the check digit d to the information string a b c. Then we want to protect the string a b c d against a single transmission error in any one of the digits a, b, c, or d. The obvious thing is to take the sum,

$$a + b + c + d = 0,$$

Suppose one digit is wrong e.g. b is misrepresented by b'. The new sum is

$$a + b' + c + d,$$

and the difference between the sums is

$$b - b'$$

which is certainly not zero.

The sum is too simple to protect us against transposition error. If, for example, a b c d becomes a d c b there is no change in the sum, so the sum check

is ineffective. There is a simple remedy - we use a weighted sum. Let's fix numbers P, Q, R, S all different and consider the check sum

$$Pa + Qb + Rc + Sd.$$

(P times a plus Q times b plus R times c plus S times d.)

This still detects single digit error provided each of P, Q, R, S is non-zero. Now consider a transposition. Because we have written everything with arbitrary letters there is no loss of generality in considering the interchange of b and d. The new check sum is

$$Pa + Qd + Rc + Sb.$$

The difference between the two check sums is

$$Q(b - d) + S(d - b) = (b - d)(Q - S).$$

We chose $Q \neq S$, so the difference is non-zero (unless, of course, $b = d$ in which case there is no change in the original message). This proves that the check sum

$$Pa + Qb + Rc + Sb$$

does everything we want.

Wait a moment! We've been working modulo 5. Does that cause any problem? For a start it cuts down the choice of the weights P, Q, R, S! These must be four different non-zero numbers and therefore they must be the numbers 1,2,3,4 in some order. That shows that the weighted check sum which we have designed is essentially the only one we could have designed!

There is another more subtle point concerning the fact that we worked modulo 5. At a crucial point in the argument we used the fact that the product of two non-zero numbers is again non-zero. Of course it is?! Actually this is true, but only because 5 is a prime number. Suppose we had been working modulo 6 then we might have produced zero by multiplying 2 times 3.

Now we have established that an effective check for the string a b c d uses the weighted sum $4a + 3b + 2c + d$. It's not hard to see that the corresponding check for a ten digit string a b c d e f g h i j should be

$$Xa + 9b + 8c + 7d + 6e + 5f + 4g + 3h + 2i + j = 0$$

and that this determines the ISBN check digit j in terms of a, b, c, d, e, f, g, h, i..

Everything will be explained once we have discussed the quick way of calculating the check digit. That should be more or less obvious by now:-

a	a	a
b	a + b	2a + b
c	a + b + c	3a + 2b + c
d	a + b + c + d	4a + 3b + 2c + d
	etc.	

The ISBN code is one simple example of an error-detecting code. Although this particular one applies to human transmission of descriptions of books, it is clear that similar problems arise in machine transmission of digitally encoded data.

Thus we have had a glimpse of the exciting new applications of pure mathematics to modern computer science.

◇ ◇ ◇ ◇