CHAPTER 12
Codes

There is a growing need for secure transmission of data. This has resulted in much research into cryptography - the art of writing in code or cipher. There are two steps involved in cryptography, first encoding the message or data and then decoding the coded message back to the original.

Substitution codes

The simplest ciphers are the substitution ones in which each letter is substituted for something else, usually a letter from the same alphabet. Here is one such example:

encoding→ ←decoding	encoding→ ←decoding
A — A	B — F
C — K	D — P
E — U	F — Z
G — E	H — J
I — 0	J — T
K — Y	L — D
M — I	N — N
O — S	P — X
Q — C	R — H
S — M	T — R
U — W	V — B
W — G	X — L
Y — Q	Z — V

The message MESSAGE would be translated into IUMMAEU.

In this cipher, the substitution for each letter was not chosen at random. If the letters of the alphabet are numbered 0 to 25 then the substitute for a letter numbered NUM is calculated as follows

NUM*5 MOD 26

The reverse process is achieved by the formula

NUM*21 MOD 26

The cipher may be listed with the following short program.

Listing 12.1
LIST

```
   10REM Simple cipher
   20MODE 6:VDU 19,0,4,0,0,0:PRINT ' TAB
(17);"Coding"'
   30FOR I=0 TO 25
   40 J=I*5 MOD 26
   50 PRINT CHR$(65+I);" - ";CHR$(65+J);
" * ";
   60NEXT
   70PRINT '' TAB(16);"Decoding"'
   80FOR I=0 TO 25
   90 J=I*21 MOD 26
  100 PRINT CHR$(65+I);" - ";CHR$(65+J);
" * ";
  110NEXT
  120PRINT ''
```

RUN

```
                  Coding

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

                 Decoding

A - A * B - V * C - Q * D - L * E - G *
F - B * G - W * H - R * I - M * J - H *
K - C * L - X * M - S * N - N * O - I *
```

```
P - D * Q - Y * R - T * S - O * T - J *
U - E * V - Z * W - U * X - P * Y - K *
Z - F *
```

Essentially to encode a letter we multiply the number of the letter by 5 and ignore multiples of 26. To decode a letter we need to divide by 5, ignoring multiples of 26. This is the same as multiplying by 21 and ignoring multiples of 26 because

```
5 * 21  = 105
        = 1 + 4 * 26
```

In other words if we ignore multiples of 26 then 21 is the reciprocal of 5. Indeed we say that 21 is the inverse of 5 modulo 26.

One obvious defect of this code is that a message such as PLEASE COME QUICKLY would be encoded as XDUAMU KSIU CWIKYDQ. Spaces are left as spaces. To overcome this objection we should include spaces, full stops, commas, digits and perhaps question marks in our list of letters for substitution.

Since we have a computer at our disposal we should use the ASCII characters. The 59 ASCII characters from 31 to 90 are convenient. These include all the letters that we require and in addition 59 is a prime number which will be useful for our purpose.

Encoding will be done essentially by multiplying by 5. More precisely the single character A$ is encoded to C$ as follows.

```
N = 5 * (ASC(A$) - 31)
C$ = CHR$(31 + (N MOD 59))
```

Decoding is achieved by multiplying by 12 (the product of 5 and 12 is 60 which is 1 ignoring multiples of 59). To decode the single character C$ we proceed as follows.

```
M = 12 * (ASC(C$) - 31)
A$ = CHR$(31 + (M MOD 59))
```

Here is a short program, based on the above cipher, which will encode a message or decode one. There is one added feature - you have to enter one of the numbers 2 to 58 for encoding and decoding. The same number must be used for decoding as for encoding. To encode we multiply by the chosen number, say N. To decode we multiply by the reciprocal of N modulo 59, that is by a number M for which N*M is 1 ignoring multiples of 59.

LIST

```
   10 REM Substitution code
   20 MODE 6:VDU 19,0,4,0,0,0:PRINT ' TA
B(12);"Substitution code"'
   30 PRINT "This prgram encodes and dec
odes."'
   40 PRINT "Enter the code number."
   50 REPEAT
   60  INPUT '"Entr number 2 to 58? " C
   70  IF C<2 OR C>58 OR C<>INT(C) THEN
PRINT '"Try a sensible number."
   80 UNTIL C>1 AND C<59 AND C=INT(C)
   90 PRINT '"Do you want to encode (E)
or decode (D) a message?"
  100 REPEAT
  110  INPUT '"E or D? " A$
  120  IF A$<>"E" AND A$<>"D" THEN PRINT
 '"Which did you say?"
  130 UNTIL A$="E" OR A$="D"
  140 D=C:I=C:B$="En"
  150 IF A$="D" THEN D=1:B$="De":REPEAT:
C=C+I:D=D+1:UNTIL C MOD 59 = 1
  160 REM D is the multiple required for
 encoding/decoding
  170 PRINT '"Type your message - up to
255 characterslong."'
  180 M$="":N$="":L=0
  190 REPEAT
  200  G=GET
  210  REM Message is entered character
by character
  220  IF G>31 AND G<91 THEN M$=M$+CHR$(
G):PRINT CHR$(G);:L=L+1
  230  IF G=127 AND L>0 THEN PRINT CHR$(
127);:L=L-1:M$=LEFT$(M$,L)
  240 UNTIL G=13 OR L>=255
  250 PRINT " "
```

```
  260 REM Encoding/decoding message
  270 FOR I=1 TO L
  280  N=ASC(MID$(M$,I,1))-31:N=D*N
  290  N$=N$+CHR$(31+(N MOD 59))
  300 NEXT
  310 PRINT '"The ";B$;"coded message is
:"''N$
  320 REM Ending
  330 COLOUR 3:PRINT CHR$(7) '' TAB(10);
"Another go? Y or N ";
  340 REPEAT:G$=GET$:UNTIL G$="Y" OR G$=
"N"
  350 IF G$="Y" THEN RUN
  360 CLS:PRINT '"Bye for now.":@%=10:EN
D
```

RUN

```
            Substitution code

This prgram encodes and decodes.

Enter the code number.

Enter number 2 to 58? 3

Do you want to encode (E) or decode (D)
a message?

E or D? E

Type your message - up to 255 characters
long.

HELLO WORLD.
```

```
The Encoded message is:

$V009"Q9B0SL


          Another go? Y or N
```

Despite all our efforts the code is easy (for experts) to break or decrypt. The problem with a substitution code is that each character is invariably represented by some other fixed character. Certain English letters and pairs of letters occur much more frequently than others. For instance, in normal English text, the letter E occurs about 13% of the time, the letter T occurs about 9%, the letter P about 2% and Q about 0.2%. Armed with such information it is possible to decrypt a substitution code. The number of different substitution codes possible using the 59 characters (ASCII codes 31 to 90) is about 1.4 * 1080. Nevertheless substitution codes can be decrypted. This illustrates how deceptive the appearance of large numbers of choices can be.

The next selection introduces codes which do not always use the same character for a given character.

Matrix codes

We can use matrices to cipher messages. We illustrate the method with an example. Suppose that we want to encode the message PLEASE COME QUICKLY. First we rearrange the message into two rows as below:

```
P E S   O E Q I K Y
L A E C M   U C L .
```

Next, create a two row matrix from these rows by converting the letters into their ASCII codes less 31.

$$\begin{bmatrix} 49 & 38 & 52 & 1 & 48 & 38 & 50 & 42 & 44 & 58 \\ 45 & 34 & 38 & 36 & 46 & 1 & 54 & 36 & 45 & 15 \end{bmatrix}$$

Now premultiply the matrix with the following matrix.

$$\begin{bmatrix} 2 & -1 \\ -5 & 3 \end{bmatrix}$$

The result is

$$\begin{bmatrix} 2 & -1 \\ -5 & 3 \end{bmatrix} * \begin{bmatrix} 49 & 38 & 52 & 1 & 48 & 38 & 50 & 42 & 44 & 58 \\ 45 & 34 & 38 & 36 & 46 & 1 & 54 & 36 & 45 & 15 \end{bmatrix}$$

$$= \begin{bmatrix} 53 & 42 & 66 & -34 & 50 & 75 & 46 & 48 & 43 & 101 \\ -110 & -88 & -146 & 103 & -102 & -187 & -88 & 102 & -85 & -245 \end{bmatrix}$$

Now convert the numbers in the matrix so that they are between 0 and 58. This is achieved by adding or subtracting multiples of 59 to the numbers in the following way:

```
N = N MOD 59 : IF N<0 THEN N = N + 59
```

The resulting matrix is:

$$\begin{bmatrix} 53 & 42 & 7 & 25 & 50 & 16 & 46 & 48 & 43 & 42 \\ 8 & 30 & 31 & 44 & 16 & 49 & 30 & 16 & 33 & 50 \end{bmatrix}$$

Finally we add 31 to these numbers and look up the corresponding characters by asking for PRINT CHR$(X). The result is shown below.

T	I	&	8	Q	/	M	O	J	I
'	=	>	K	/	P	=	/	@	Q

The final coded message is

```
T'i=&>8KQ//PM=O/J@IQ
```

Notice that the letter E appears three times in the original message. These three Es have been encoded to I, > and /. Thus this cipher is more subtle than the straight substitution code.

A code is no good unless we can decipher the messages. The trick now is to use the reverse process. At first sight this may look difficult, but we use some mathematics to help us.

Let's decipher the following message.

```
I=M?"S?$:8M@>'<M
```

First we write the coded message in two rows.

I	M	"	?	:	M	>	<
=	?	S	$	8	@	'	M

Next, calculate the ASCII codes less than 31 to produce a matrix.

$$\begin{bmatrix} 42 & 46 & 3 & 32 & 27 & 46 & 31 & 29 \\ 30 & 32 & 52 & 5 & 25 & 33 & 8 & 46 \end{bmatrix}$$

Now premultiply by the following matrix.

$$\begin{bmatrix} 3 & 1 \\ 5 & 2 \end{bmatrix}$$

Notice that this matrix is not the same as the one used for encoding. However, notice also that the decoding matrix is the inverse of the encoding matrix as the following calculation shows.

$$\begin{bmatrix} 3 & 1 \\ 5 & 2 \end{bmatrix} * \begin{bmatrix} 2 & -1 \\ -5 & 3 \end{bmatrix}$$

$$= \begin{bmatrix} 3*2 + 1*-5 & 3*-1 + 1*3 \\ 5*2 + 2*-5 & 5*-1 + 2*3 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Let' s multiply our decoding matrix by the coded message matrix.

$$\begin{bmatrix} 3 & 1 \\ 5 & 2 \end{bmatrix} * \begin{bmatrix} 42 & 46 & 3 & 32 & 27 & 46 & 31 & 29 \\ 30 & 32 & 52 & 5 & 25 & 33 & 8 & 46 \end{bmatrix}$$

$$\begin{bmatrix} 156 & 170 & 61 & 101 & 106 & 171 & 101 & 133 \\ 270 & 294 & 119 & 170 & 185 & 296 & 171 & 237 \end{bmatrix}$$

Next, add (or subtract) multiples of 59 to make the numbers between 0 and 58.

$$\begin{bmatrix} 38 & 52 & 2 & 42 & 47 & 53 & 42 & 15 \\ 34 & 58 & 1 & 52 & 8 & 1 & 53 & 1 \end{bmatrix}$$

Now add 31 to these numbers and look up the corresponding characters.

E	S	!	I	N	T	I.
A	Y		S	'		T

We thus end up with the message EASY! ISN' T IT.

The next program uses such matrices to code and/or decode messages.

Listing 12.3
LIST

```
   10 REM Matrix cipher
   20 MODE 6:VDU 19,0,4,0,0,0:PRINT ' TA
B(13);"Matrix cipher"'
   30 PRINT "This program encodes and de
codes."
   40 S=2:DIM A(S,S):FOR I=1 TO S:FOR J=
1 TO S:READ A(I,J):NEXT:NEXT
   50 PRINT '"Do you want to encode (E)
or decode (D) a message?"
   60 REPEAT
   70  INPUT '"E or D? " A$
   80  IF A$<>"E" AND A$<>"D" THEN PRINT
 '"Which did you say?"
   90 UNTIL A$="E" OR A$="D"
  100 B$="En"
  110 IF A$="D" THEN B$="De":FOR I=1 TO
S:FOR J=1 TO S:READ A(I,J):NEXT:NEXT
  120 PRINT '"Type your message - up to
255 characterslong."'
  130 M$="":N$="":L=0
  140 REPEAT
  150  G=GET
  160  REM Message is entered character
by character
  170  IF G>31 AND G<91 THEN M$=M$+CHR$(
G):PRINT CHR$(G);:L=L+1
  180  IF G=127 AND L>0 THEN PRINT CHR$(
127);:L=L-1:M$=LEFT$(M$,L)
  190 UNTIL G=13 OR L>=255
  200 PRINT " ":M=INT(L/S+0.9):DIM B(S,M
),C(S,M)
  210 REM Encoding/decoding message
  220 FOR I=1 TO S:FOR J=1 TO M
  230  K=S*J+I-S
```

```
  240  IF K<=L THEN N=ASC(MID$(M$,K,1))-
31:B(I,J)=N MOD 59
  250 NEXT:NEXT
  260 FOR I=2 TO S:IF B(I,M)<1 THEN B(I,
M)=1
  270 NEXT
  280 FOR I=1 TO S:FOR J=1 TO M
  290  FOR K=1 TO S:C(I,J)=C(I,J)+A(I,K)
*B(K,J):NEXT
  300  C(I,J)=C(I,J) MOD 59:IF C(I,J)<0
THEN C(I,J)=C(I,J)+59
  310 NEXT:NEXT
  320 PRINT '"The ";B$;"coded message is
:"'
  330 FOR J=1 TO M:FOR I=1 TO S
  340  PRINT CHR$(31+C(I,J));
  350 NEXT:NEXT
  360 REM Ending
  370 COLOUR 3:PRINT CHR$(7) '' TAB(10);
"Another go? Y or N ";
  380 REPEAT:G$=GET$:UNTIL G$="Y" OR G$=
"N"
  390 IF G$="Y" THEN RUN
  400 CLS:PRINT '"Bye for now.":@%=10:EN

  410 DATA 2,-1,-5,3
  420 DATA 3,1,5,2
```

RUN

```
             Matrix cipher

This program encodes and decodes.

Do you want to encode (E) or decode (D)
a message?

E or D? E
```

```
Type your message - up to 255 characters
long.

HELLO WORLD.

The Encoded message is:

K:L;CY$HXXCLS

            Another go? Y or N
```

For this cipher we used a two by two matrix to encode and the inverse of the matrix to decode. In general we can use any matrix and its inverse as long as both matrices have only integers as their entries. Here is another matrix that you could use for encoding.

$$\begin{bmatrix} 0 & 1 \\ 1 & -2 \end{bmatrix}$$

The corresponding decoding matrix is given next.

$$\begin{bmatrix} 2 & 1 \\ 1 & 0 \end{bmatrix}$$

This method of coding could be made more suitable by using 3 by 3 matrices. The message would need to be written out in three rows and the procedure outlined earlier on followed. Here are two matrices that could be used as encoders and decoders.

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 1 & 3 \\ 4 & 2 & 5 \end{bmatrix} \qquad \begin{bmatrix} 1 & 2 & -1 \\ -2 & -9 & 5 \\ 0 & 2 & -1 \end{bmatrix}$$

The next program uses these matrices to code and/or decode messages. You could insert your own matrices if you wish, but make sure that they are inverse to each other and that all entries are integral.

LIST

```
   10 REM Matrix cipher
   20 MODE 6:VDU 19,0,4,0,0,0:PRINT ' TA
B(13);"Matrix cipher"'
```

```
   30 PRINT "This program encodes and de
codes."
   40 S=3:DIM A(S,S):FOR I=1 TO S:FOR J=
1 TO S:READ A(I,J):NEXT:NEXT
   50 PRINT '"Do you want to encode (E)
or decode (D) a message?"
   60 REPEAT
   70  INPUT '"E or D? " A$
   80  IF A$<>"E" AND A$<>"D" THEN PRINT
 '"Which did you say?"
   90 UNTIL A$="E" OR A$="D"
  100 B$="En"
  110 IF A$="D" THEN B$="De":FOR I=1 TO
S:FOR J=1 TO S:READ A(I,J):NEXT:NEXT
  120 PRINT '"Type your message - up to
255 characterslong."'
  130 M$="":N$="":L=0
  140 REPEAT
  150  G=GET
  160  REM Message is entered character
by character
  170  IF G>31 AND G<91 THEN M$=M$+CHR$(
G):PRINT CHR$(G);:L=L+1
  180  IF G=127 AND L>0 THEN PRINT CHR$(
127);:L=L-1:M$=LEFT$(M$,L)
  190 UNTIL G=13 OR L>=255
  200 PRINT " ":M=INT(L/S+0.9):DIM B(S,M
),C(S,M)
  210 REM Encoding/decoding message
  220 FOR I=1 TO S:FOR J=1 TO M
  230  K=S*J+I-S
  240  IF K<=L THEN N=ASC(MID$(M$,K,1))-
31:B(I,J)=N MOD 59
  250 NEXT:NEXT
  260 FOR I=2 TO S:IF B(I,M)<1 THEN B(I,
M)=1
  270 NEXT
  280 FOR I=1 TO S:FOR J=1 TO M
```

```
  290  FOR K=1 TO S:C(I,J)=C(I,J)+A(I,K)
*B(K,J):NEXT
  300  C(I,J)=C(I,J) MOD 59:IF C(I,J)<0
THEN C(I,J)=C(I,J)+59
  310 NEXT:NEXT
  320 PRINT '"The ";B$;"coded message is
:"'
  330 FOR J=1 TO M:FOR I=1 TO S
  340  PRINT CHR$(31+C(I,J));
  350 NEXT:NEXT
  360 REM Ending
  370 COLOUR 3:PRINT CHR$(7) '' TAB(10);
"Another go? Y or N ";
  380 REPEAT:G$=GET$:UNTIL G$="Y" OR G$=
"N"
  390 IF G$="Y" THEN RUN
  400 CLS:PRINT '"Bye for now.":@%=10:EN
D
  410 DATA 1,0,-1,2,1,3,4,2,5
  420 DATA 1,2,-1,-2,-9,5,0,2,-1
```

RUN

```
             Matrix cipher

This program encodes and decodes.

Do you want to encode (E) or decode (D)
a message?

E or D? E

Type your message - up to 255 characters
long.

HELLO WORLD

The Encoded message is:
```

```
V2SK6L$1K=UA

        Another go? Y or N
```

Public-key codes

The codes described in the previous section have a defect. Once you know how to encode a message you also know how to decode it. Public-key cryptosystems are different. They come in two parts: the encoding key, which is made public, enabling anyone to encode messages; and the decoding key, which is kept secret, enabling only the originator of the code to decode messages.

We now describe a public-key system. First find two very large prime numbers P and Q: each should have about 50 decimal digits making this proposition impractical for your computer. Let N be the product of P and Q. Now choose an integer A which is less than N and has no factor common with (P - 1)*(Q - 1). You may now publicly announce the numbers N and A.

How is a message encoded with the numbers N and A? This is performed as follows.

1. Translate the message into numbers (space = 01, A = 34, etc.); the message is then one large number.
2. Take the message in number form and break it up into blocks of a convenient size.
3. Encode each block B as follows:

   C = B↑A MOD N

   the number B is raised to the power A and multiples of N are removed to make the resulting number between 0 and N.

How is the resulting message decoded? Since the greatest common divisor of A and (P - 1)*(Q - 1) is 1 we can find two numbers X and Y so that A*X + (P - 1)*(Q - 1)*Y = 1. Using X we can decode the message:

1. Break the coded message into blocks
2. For each block C perform the following calculation:

   C↑X MOD N

3. Join the resulting blocks back again and decode numbers back to characters.

The process works because

$$(B\uparrow A)\uparrow X = B\uparrow(A*X)$$
$$= B\uparrow(1 - (P - 1)*(Q - 1)*Y)$$
$$= B + \text{multiples of N.}$$

The last statement follows from a theorem proved by the mathematician Fermat.

Why is the code difficult to break? Notice that to decode the message we need to know the value of X. This can be calculated from the value of (P - 1)*(Q - 1). In order to know (P - 1)*(Q - 1) we need to know P and Q. Publicly only A and N have been announced. In theory, once we know N we can factorise it to find P and Q. However N is about 100 decimal digits long and factorisation of such numbers takes an enormous amount of computer time to perform (several millions of years). It is on this basis that the cipher is safe.

All that remains for you to produce a very secure code is to write a program for your BBC or Electron which is capable of handling very long numbers precisely. See the chapter entitled ' Odds and Ends' .