SECRET CODES

BY

JOHN LOXTON

The art of writing secret messages, intelligible to those who are in possession of the key and unintelligible to all others, has been studied for centuries. Military gentlemen and spies depend on secret codes for their very existence. On the other hand, every self-respecting Ministry of Information employs legions of rather pale people with eye-shades to decipher messages intercepted from the enemy. The computer revolution has had an enormous impact on this world. Since the coding and decoding no longer have to be done by hand, much more complicated schemes can be used. The computer has also created new uses for secret codes. In the not so distant future, some sort of electronic message may be the only record of a contract or a business transaction. Now a bit of memory in a computer cannot be signed, sealed and witnessed in the same way as a piece of parchment. How can the electronic record be stored so that its authenticity can be verified and so that no-one can change the contents of the record without proper authorisation? This series of articles will explore secret codes, both ancient and modern, and describe some of the recent developments in this fascinating subject.

The essential feature of the secret codes considered here is that the information they contain should remain hidden from anyone who might obtain copies of the messages transmitted but who does not know the key. The ancient Romans are said to have communicated secret messages by shaving a slave's head, inscribing the message on his scalp and then sending the slave to deliver the message after his hair had grown back again. This is ingenious, but it is not a secret code. A method for converting a plain-text message into a secret message has two basic parts, namely the encryption system which is fixed, and the key which may vary from message to message. It is reasonable to assume that the enemy has full knowledge of the encryption system because this will be used over a long period and will be known to many secret communicators. The security of the system depends on the difficulty in discovering the key from secret messages that happen to be intercepted. One thing that works in favour of the cryptanalyst is that he will usually have many messages enciphered using the same key, because the key cannot be changed too often without confusing the intended recipient of the message.

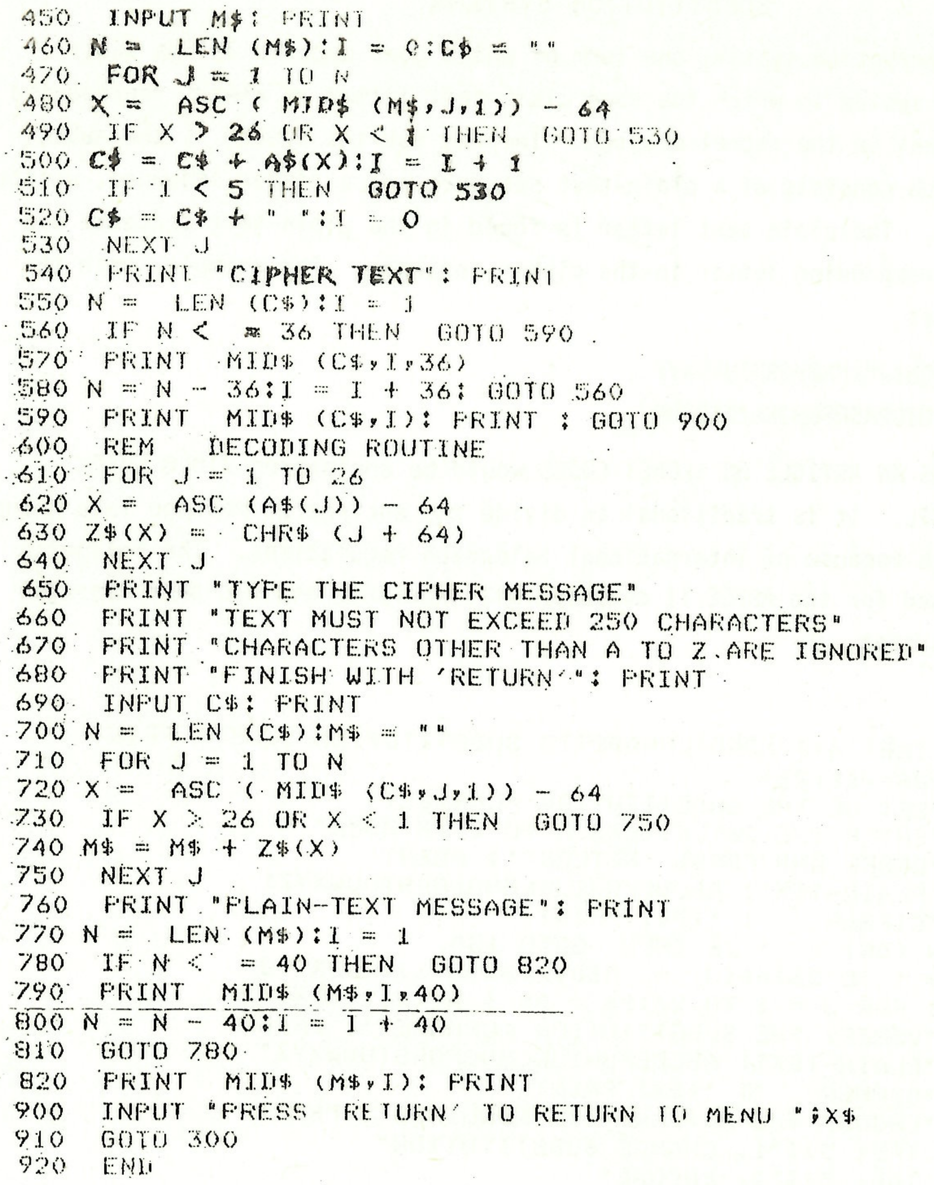
SUBSTITUTION SYSTEMS

The simplest encryption system, one form of which goes back to Julius Caesar, is the substitution system in which the same plain-text letter is always represented by the same equivalent in the secret message. The key for the system is the substitution alphabet which consists of a plain-text sequence and a cipher sequence, written one above the other. The plain-text letter is found in the plain-text sequence and replaced by the corresponding letter in the cipher sequence. For example, with the substitution alphabet

```
Plain text    ABCDEFGHIJKLMNOPQRSTUVWXYZ
Cipher        QWERTYUIOPASDFGHJKLZXCVBNM
```

the sentence THIS IS AN ARTICLE ON SECRET CODES would be enciphered ZIOLO LQFQK ZOEST GFLTE KTZEG RTL. It is traditional to divide the enciphered message into groups of five letters each because of international telegraph regulations. The following BASIC program devised for the APPLE II computer will encipher and decipher messages according to this system.

```
 100  HOME
 110  PRINT  TAB( 4);"MONOALPHABETIC SUBSTITUTION CODE": PRINT
 120  DIM A$(26),Z$(26)
 130  PRINT "SET UP THE SUBSTITUTION ALPHABET"
 140  PRINT "ENTER THE 26 LETTERS A TO Z IN SOME"
 150  PRINT "ORDER AND PRESS 'RETURN'": PRINT
 160  PRINT "PLAIN-TEXT: ABCDEFGHIJKLMNOPQRSTUVWXYZ"
 170  INPUT "CIPHER     : ";B$: PRINT
 180  IF  LEN (B$) <  > 26 THEN  GOTO 130
 190  FOR J = 1 TO 26:A$(J) =  MID$ (B$,J,1): NEXT J
 200 B$ = "": FOR J = 1 TO 26:B$ = B$ + A$(J): NEXT J
 210  PRINT "VERIFY THE SUBSTITUTION ALPHABET": PRINT
 220  PRINT "PLAIN-TEXT: ABCDEFGHIJKLMNOPQRSTUVWXYZ"
 230  PRINT "CIPHER     : ";B$: PRINT
 300  PRINT "CHOOSE THE ALTERNATIVE REQUIRED": PRINT
 310  PRINT  TAB( 5);"1. CHANGE SUBSTITUTION"
 320  PRINT  TAB( 5);"2. ENCODE"
 330  PRINT  TAB( 5);"3. DECODE"
 340  PRINT  TAB( 5);"4. EXIT"
 350  INPUT "ENTER 1,2,3 OR 4 AND PRESS RETURN ";I: HOME
 360  IF I = 1 THEN  GOTO 130
 370  IF I = 2 THEN  GOTO 410
 380  IF I = 3 THEN  GOTO 610
 390  GOTO 920
 400  REM   ENCODING ROUTINE
 410  PRINT "TYPE PLAIN-TEXT MESSAGE"
 420  PRINT "TEXT MUST NOT EXCEED 250 CHARACTERS"
 430  PRINT "CHARACTERS OTHER THAN A TO Z ARE IGNORED"
 440  PRINT "FINISH WITH 'RETURN'": PRINT
```



```
450  INPUT M$: PRINT
460 N =  LEN (M$):I = 0:C$ = ""
470  FOR J = 1 TO N
480 X =  ASC ( MID$ (M$,J,1)) - 64
490  IF X > 26 OR X < 1 THEN  GOTO 530
500 C$ = C$ + A$(X):I = I + 1
510  IF I < 5 THEN  GOTO 530
520 C$ = C$ + " ":I = 0
530  NEXT J
540  PRINT "CIPHER TEXT": PRINT
550 N =  LEN (C$):I = 1
560  IF N <  = 36 THEN  GOTO 590
570  PRINT  MID$ (C$,I,36)
580 N = N - 36:I = I + 36: GOTO 560
590  PRINT  MID$ (C$,I): PRINT : GOTO 900
600  REM   DECODING ROUTINE
610  FOR J = 1 TO 26
620 X =  ASC (A$(J)) - 64
630 Z$(X) =  CHR$ (J + 64)
640  NEXT J
650  PRINT "TYPE THE CIPHER MESSAGE"
660  PRINT "TEXT MUST NOT EXCEED 250 CHARACTERS"
670  PRINT "CHARACTERS OTHER THAN A TO Z ARE IGNORED"
680  PRINT "FINISH WITH 'RETURN'": PRINT
690  INPUT C$: PRINT
700 N =  LEN (C$):M$ = ""
710  FOR J = 1 TO N
720 X =  ASC ( MID$ (C$,J,1)) - 64
730  IF X > 26 OR X < 1 THEN  GOTO 750
740 M$ = M$ + Z$(X)
750  NEXT J
760  PRINT "PLAIN-TEXT MESSAGE": PRINT
770 N =  LEN (M$):I = 1
780  IF N <  = 40 THEN  GOTO 820
790  PRINT  MID$ (M$,I,40)
800 N = N - 40:I = I + 40
810  GOTO 780
820  PRINT  MID$ (M$,I): PRINT
900  INPUT "PRESS  RETURN' TO RETURN TO MENU ";X$
910  GOTO 300
920  END
```

Although there are an enormous number of different substitutions, in fact 26! ≈ 10^{26}, the simple substitution cipher is fairly easy to crack. The method is based on the relative frequencies of the individual letters of the alphabet in standard English text, together with the relative frequencies of their combinations with each other In English, the relative frequencies of occurence of the individual letters and the most frequent digraphs and trigraphs are as follows

Letter	Frequency	Letter	Frequency	Digraph	Frequency	Trigraph	Frequency
E	.131	U	.028	TH	.034	THE	.015
T	.090	M	.026	HE	.026	AND	.005
O	.082	P	.022	AN	.019	THA	.004
A	.078	Y	.015	ER	.019	HAT	.003
N	.073	W	.015	ON	.019	EDT	.003
I	.068	G	.014	RE	.017	ENT	.003
R	.067	B	.013	IN	.014	FOR	.003
S	.065	V	.010	ED	.014	ION	.003
H	.059	K	.004	ND	.014	TIO	.003
D	.044	X	.003	AT	.013	NDE	.003
L	.036	J	.001	OF	.013	HAS	.002
C	.029	Q	.001	OR	.012	MEN	.002
F	.028	Z	.001	HA	.012		
				EN	.011		
				NT	.011		

Once a few identifications have been made, by identifying the high frequency letters or detecting common word patterns, a complete solution follows readily. Usually, the substitution can be cracked from about 25 to 50 characters of the cipher text. The following program will perform the rather tedious letter counts, but human intervention is required to make the identifications.

```
100  HOME
110  PRINT "  MONOALPHABETIC SUBSTITUTION SOLVER": PRINT
120  DIM A(26),B(26,26)
130  PRINT "TYPE THE CIPHER MESSAGE"
140  PRINT "TEXT MUST NOT EXCEED 250 CHARACTERS"
150  PRINT "CHARACTERS OTHER THAN A TO Z ARE IGNORED"
160  PRINT "FINISH WITH 'RETURN'": PRINT
170  INPUT C$:N =  LEN (C$)
180  FOR J = 1 TO 26:A(J) = 0: FOR K = 1 TO 26:B(J,K) = 0: NEXT K: NEXT J
190  FOR J = 1 TO N:X =  ASC ( MID$ (C$,J,1)) - 64
195  IF X > 0 AND X < 27 THEN  GOTO 200: NEXT J
200 A(X) = A(X) + 1:I = 1
210  FOR J = 2 TO N
220 Y =  ASC ( MID$ (C$,J,1)) - 64
230  IF Y < 1 OR Y > 26 THEN  GOTO 250
240 A(Y) = A(Y) + 1:B(X,Y) = B(X,Y) + 1:I = I + 1:X = Y
250  NEXT J
260  FOR J = 1 TO 26:A(J) =  INT (100 * A(J) / I + 0.5): FOR K = 1 TO 26
265 B(J,K) =  INT (100 * B(J,K) / (I - 1) + 0.5): NEXT K: NEXT J
270  HOME : PRINT "LETTER FREQUENCIES (%)": PRINT
280 D$ = ""
290  FOR J = 1 TO 26
300 X$ =  STR$ (A(J)):N =  LEN (X$)
```

```
310  IF N = 1 THEN  GOTO 330
320 D$ = D$ + X$ + " '; GOTO 340
330 D$ = D$ + " " + X$ + " "
340  NEXT J                                                              [PRINT
350  PRINT " A  B  C  D  E  F  G  H  I  J  K  L  M": PRINT  LEFT$ (D$,39):
360  PRINT " N  O  P  Q  R  S  T  U  V  W  X  Y  Z": PRINT  RIGHT$ (D$,39):
370  PRINT "COMMON DIGRAPHS (%)": PRINT                                  [PRINT
380 D$ = "":I = 0
390  FOR J = 1 TO 26
400  FOR K = 1 TO 26
410  IF B(J,K) < 2 THEN  GOTO 450
420 X$ =  STR$ (B(J,K));N =  LEN (X$)
430  IF N = 1 THEN X$ = " " + X$
440 D$ = D$ +  CHR$ (J + 64) +  CHR$ (K + 64) +. " " + X$ + "  ":I = I + 1
450  NEXT K
460  NEXT J
470 J = 1
480  IF I < 6 THEN  GOTO 500
490. PRINT  MID$ (D$,J,35):I = I - 5:J = J + 35: GOTO 480
500  PRINT  MID$ (D$,J)
510  END
```

There are some secret messages for you to try to crack
(Solutions are on page 23.)

(1) GPJKP VRPGG PJKGO HGDCG OKSTK CGDPR
UOKGO KVGDC RPJLK VDRGO KEDRZ GPCTA
AKVGO KCLDR NCHRZ HVVPU CPAPT GVHNK
PTCAP VGTRK PVGPG HWKTI HVECH NHDRC
GHCKH PAGVP TJLKC HRZJF PIIPC DRNKR
ZGOKE

(2) UIEHR PUEPU QPAAE GUIRL UVEQU JEPUR
LUIWX HRCVX EQXXG EHWLW EAWPU IEXNF
FQYCX QPAPR UEXWP AWHQU EUIQU WUWXW
PUEPA EALRV XUCAE PUXSI RIQOE HRJGF
EUEAU IEXHI RRFHE VUWLW HQUEJ QUIEJ
QUWHX HRCVX E

(3) EKUQN UREKU PCHXN UQTEK UKVIQ EUTHP
UAPUC HXMEQ EKUPH RQWEK UPCHX NUPQT
EKUQE KUNEB QPAYU P

Readers are invited to improve the computer programs in this article. Can you automate more of the solution procedure?

What about other codes? Write and tell us about any interesting codes you have come across. Further articles on this subject are being planned, going on to describe some recent exciting developments in coding theory.