APPENDIX F

CRYPTANALYSIS SUPPORT PROGRAM

F-1. Program Support

This program supports the development of FM 34-40-2, Basic Cryptanalysis. It gives the capability to encipher and decipher messages in monoalphabetic and polyalphabetic substitution systems, produce a variety of statistical data about the encrypted messages, and print the results or save them to disk. Because of its limited purpose, the program does not support on-screen analysis. The printed results can be used off-line to aid in analysis, however. The program should be particularly useful in preparing examples and exercises for training cryptanalytic techniques.

F-2. On-screen Analysis

The logical structure is present in the program to support on-screen analysis, if desired. The coding that now sends results to disk or printer can be modified to display on screen as well. Lines 6060 through 6780 provide the basis for this. This code together with the alphabet entry subroutines in lines 3920 through 5760 can be used to enter partial trial recoveries and see the results for both monoalphabetic and polyalphabetic systems.

F-3. Program Format

The listing has been specially formatted to make it easy to follow the program logic. Each statement in multiple statement numbered lines has been printed on a separate line with each follow-on statement indicated by the statement separator (colon) at the beginning of the line. FOR-NEXT commands have been indented to show the level and structure of each. Similarly, the parts of IF...THEN...ELSE statements have been printed on separate lines and then indented to show their structure clearly. If the program is typed in by hand, the statements in a single numbered line should be entered continuously, not on separate lines in most versions of BASIC. Indentation of FOR-NEXT structures can be preserved, if desired, but not for IF...THEN...ELSE statements.

```
100   ' CRYPTANALYSIS SUPPORT PROGRAM
120   ' Version 1.0
140   ' 4 October 1988
160   '
180   ' Developed in support of FM 34-40-2, Basic Cryptanalysis to provide
200   ' accurate encryption, decryption, frequency counts, and statistics for use
220   ' in the manual. It can be used for other applications.
240   '
260   ' The program was written in GW-BASIC.
280   ' It is readily adaptable to any computers that run
300   ' GW-BASIC. It can easily be converted to run in other BASIC languages.
320   '
340   ' As written, the program will print on a dot matrix printer with the name
360   ' PRN1 that uses standard Epson control codes. If necessary, change the
380   ' values in the *** Printer Setup *** section for the particular printer
400   ' to be used.
420   '
440   ' *** Printer Setup ***
460   PRINTER$=“PRN1”
480   FORMFEED$=CHR$(12)
500   CRLF$=CHR$(13)+CHR$(10) ' (not used in 1.0)
520   CONDENSED$=CHR$(15) ' (not used in 1.0)
540   DC2$=CHR$(18) ' Cancels condensed mode (not used in 1.0)
560   ELITE$=CHR$(27)+“M” ' (not used in 1.0)
580   PICA$=CHR$(27)+“P” ' (not used in 1.0)
600   '
620   ' *** Initialize Variables ***
640   DIM PTEXTD$(25), PTEXTI$(25), CTEXTD$(25), CTEXTI$(25)
660   ' Plain and ciphertext may be stored in two forms: display and internal.
680   ' Display forms (PTEXTD$() and CTEXTD()) are as typed with spaces.
700   ' Internal forms (PTEXTI$() and CTEXTI$()) have spaces, and nonliteral
720   ' characters stripped away. All frequency counts and ICs are performed on
740   ' CTEXTI$() strings. Up to 25 lines of text are allowed, as written.
760   ' Additional lines of text may be used if all uses of “25” are increased
780   ' in the DIM statement in line 640.
800   DIM MFREQ(26), PFREQ(20,27), DIFREQ(26,26), PHIMONO,PHIPERI(20), PHIDIG,
      PMIXFREQ(20,27), SET 1(26), SET 2(27), MATCH (27), PERPHISUM(20), PERTOTLTR(20)
820   ' Sets up monographic, periodic, and digraphic frequency, IC tables. Up
840   ' to 20 alphabets are allowed for periodic frequencies, as written. The
860   ' number of alphabets can be increased by increasing all uses of “20” in
880   ' the DIM statements in line 800.
900   DIM PCOMP$, CCOMP$(200) ' Variables for plain and cipher components with up
920   ' to 200 cipher component sequences for long running key aperiodics. The
940   ' length of the key may be increased by increasing the “200” in the DIM
960   ' statement in line 900.
1000  '
1020  KEY OFF ' Turns off prompts on bottom of screen.
1040  '
```

```
1160  ' *** Main Menu ***
1180  CLS
1200  PRINT "          CRYPTANALYSIS SUPPORT PROGRAM"
1220  PRINT
     :PRINT
1240  PRINT "      1. Enter text ";STATUS$(1)
1260  PRINT "      2. Encipher text ";STATUS$(2)
1280  PRINT "      3. Decipher text ";STATUS$(3)
1300  PRINT "      4. Print text ";STATUS$(4)
1320  PRINT "      5. Save text to disk ";STATUS$(5)
1340  PRINT "      6. Calculate frequency counts, ICs ";STATUS$(6)
1360  PRINT "      7. Print frequency counts, ICs ";STATUS$(7)
1380  PRINT "      8. Save frequency counts, ICs to disk ";STATUS$(8)
1400  PRINT "      9. Find repeats ";STATUS$(9)
1420  PRINT "     10. Quit"
1440  PRINT
     :PRINT
1460  '
1480  ' *** Main Menu Control ***
1500  INPUT  " Enter your choice: ",SELECTION
1520  ON SELECTION GOSUB 1600,3000,3480,6080,6380,6840,8600,9960,10240,10980
1540  GOTO 1180
1560  '
1580  ' *** Text Entry Subroutine ***
1600  CLS
1620  PRINT "          TEXT ENTRY MENU"
1640  PRINT
     :PRINT
     :PRINT
1660  PRINT "     1. Enter plaintext from disk
1680  PRINT "     2. Enter ciphertext from disk
1700  PRINT "     3. Enter plaintext from keyboard
1720  PRINT "     4. Enter ciphertext from keyboard
1740  PRINT "     5. Return to Main Menu
1760  PRINT
     :PRINT
1780  INPUT "Enter your choice:   ", CHOICE
1800  ON CHOICE GOTO 1860,2040,2220,2440,2600
1820  '
1840  ' *** Plaintext Disk Entry ***
1860  INPUT "Enter input filename, for example, A:SAMPLE.TXT    ",INFILE$
1880  OPEN INFILE$ FOR INPUT AS #1
1900  NRLINES=0
1920  NRLINES=NRLINES+1
1940  INPUT #1, PTEXTD$(NRLINES)
1960  IF EOF(1)
        THEN STATUS$(1)="     (PLAINTEXT ENTERED)"
       :CLOSE #1
       :RETURN
```

```
1980  GOTO 1920
2000  '
2020  ' *** Ciphertext Disk Entry ***
2040  INPUT "Enter input filename, for example, A:SAMPLE.TXT   ",INFILE$
2060  OPEN INFILE$ FOR INPUT AS #1
2080  NRLINES=0
2100  NRLINES=NRLINES+1
2120  INPUT #1,CTEXTD$(NRLINES)
2140  IF EOF(1)
        THEN CLOSE #1
       :STATUS$="     (CIPHERTEXT ENTERED)"
       :GOTO 2660 ' Branches to internal text preparation.
2160  GOTO 2100
2180  '
2200  ' *** Plaintext Keyboard Entry ***
2220  PRINT "Type a line of text. Use lower case letters only."
2240  PRINT "Use no commas in the text. When you are through,"
2260  PRINT "type END on a new line."
2280  NRLINES=0
2300  LINE INPUT T$
2320  IF T$="END" OR T$="end"
        THEN STATUS$(1)="     (PLAINTEXT ENTERED)"
       :RETURN
2340  NRLINES=NRLINES+1
2360  PTEXTD$(NRLINES)=T$
2380  GOTO 2300
2400  '
2420  ' *** Ciphertext Keyboard Entry ***
2440  PRINT "Type a line of text. Use CAPITAL letters only."
2460  PRINT "When you are through, type END on a new line."
2480  NRLINES=0
2500  INPUT T$
2520  IF T$="END" OR T$="end"
        THEN STATUS$(1)="     (CIPHERTEXT ENTERED)"
       :GOTO 2660
2540  NRLINES=NRLINES+1
2560  CTEXTD$(NRLINES)=T$
2580  GOTO 2500
2600  RETURN
2620  '
2640  ' *** Preps Ciphertext in Internal Format ***
2660  FOR TEXTLINE=1 TO NRLINES
2680       T$=CTEXTD$(TEXTLINE)
2700       POSN=0
2720       POSN=POSN+1
          :IF POSN>LEN(T$)
             THEN 2800
2740       C$=MID$(T$,POSN,1)
```

```
2760        IF (ASC(C$)<65 OR ASC(C$)>90) AND C$<>“.”
              THEN GOSUB 2900
2780        GOTO 2720
2800        CTEXTI$(TEXTLINE)=T$
2820   NEXT TEXTLINE
2840   RETURN
2860   '
2880   ' *** Subroutine to Strip Nonliteral Characters From Ciphertext ***
2900   T$=MID$(T$,1,POSN-1)+MID$(T$,POSN+1,LEN(T$)-POSN)
2920   POSN=POSN-1
2940   RETURN
2960   '
2980   ' *** Encipherment Subroutine ***
3000   GOSUB 3940
3020   CYCLEPOS=0
3040   FOR LNE=1 TO NRLINES
         :CTEXTD$(LNE)=”
         :KTEXTD$(LNE)=”
       :NEXT LNE
3060   FOR LNE=1 TO NRLINES
3080     FOR CHARPOS=1 TO LEN(PTEXTD$(LNE))
3100        PCHAR$=MID$(PTEXTD$(LNE),CHARPOS,1)
3120        IF PCHAR$=" "
              THEN CCHAR$=" "
              :KCHAR$=" "
              :GOTO 3320
3140        CYCLEPOS=CYCLEPOS+1
           :IF CYCLEPOS>PERIOD
              THEN CYCLEPOS=1
3160        KCHAR$=MID$(REPEATKEY$,CYCLEPOS,1)
3180        IF ASC (PCHAR$) >64 AND ASC(PCHAR$)<91
                THEN PCHAR$=CHR$(ASC(PCHAR$)+32)
3200        IF ASC(PCHAR$)<97 OR ASC(PCHAR$)>122
                THEN PCHAR$=“.”
3220        IF PCHAR$=“.”
                THEN CCHAR$=“.”
               :GOTO 3320
3240        FOR ALPHCHAR=1 TO 26
3260           IF PCHAR$=MID$(PCOMP$,ALPHCHAR,1)
                THEN CCHAR$=MID$(CCOMP$(CYCLEPOS),ALPHCHAR,1)
               :GOTO 3320
3280        NEXT ALPHCHAR
3300        CCHAR$=“.”
3320        CTEXTD$(LNE)=CTEXTD$(LNE)+CCHAR$
          :KTEXTD$(LNE)=KTEXTD$(LNE)+KCHAR$
3340     NEXT CHARPOS
3360   NEXT LNE
3380   GOSUB 2660
```

```
3400  STATUS$(2)=“      (ENCIPHEREMENT COMPLETED)”
3420  RETURN
3440  '
3460  ' *** Decipherment Subroutine ***
3480  GOSUB 3940
3500  CYCLEPOS=0
3520  FOR LNE=1 TO NRLINES
        :PTEXTD$(LNE)=" ":
      NEXT LNE
3540  FOR LNE=1 TO NRLINES
3560    FOR CHARPOS=1 TO LEN(CTEXTD$(LNE))
3580      CCHAR$=MID$(CTEXTD$(LNE),CHARPOS,1)
3600      IF CCHAR$=“  ”
            THEN PCHAR$=“  ”
            :GOTO 3780
3620      CYCLEPOS=CYCLEPOS+1:
          IF CYCLEPOS>PERIOD
            THEN CYCLEPOS=1
3640      IF ASC(CCHAR$)>96 AND ASC(CCHAR$)<123
            THEN CCHAR$=CHR$(ASC(CCHAR$)-32)
3660      IF ASC(CCHAR$)<65 OR ASC(CCHAR$)>96
            THEN CCHAR$=“.”
3680      IF CCHAR$=“.”
            THEN PCHAR$=“.”
            :GOTO 3780
3700      FOR ALPHCHAR=1 TO 26
3720        IF CCHAR$=MID$(CCOMP$(CYCLEPOS),ALPHCHAR,1)
              THEN PCHAR$=MID$(PCOMP$,ALPHCHAR,1)
              :GOTO 3780
3740      NEXT ALPHCHAR
3760      PCHAR$=“.”
3780      PTEXTD$(LNE)=PTEXTD$(LNE)+PCHAR$
3800    NEXT CHARPOS
3820  NEXT LNE
3840  GOSUB 2660
3860  STATUS$(3)=“      (DECIPHERMENT COMPLETED)”
3880  RETURN
3900  '
3920  ' *** Alphabet Entry Subroutine ***
3940  PCOMP$=“abcdefghijklmnopqrstuvwxyz”
3960  CCOMPO$=“ABCDEFGHIJKLMNOPQRSTUVWXYZ”
3980  RKEY$=“AAAAAAAAAAAAAAAAAAAA”
4000  PERIOD=1
4020  CLS
4040  PRINT “Select type of system:”
     :PRINT
4060  PRINT “     1. Monoalphabetic uniliteral”
4080  PRINT “     2. Periodic polyalphabetic”
4100  PRINT “     3. Aperiodic polyalphabetic”
```

```
4120  PRINT
     :PRINT
4140  INPUT “Enter your choice:   ”, SELECTION
4160  ON SELECTION GOSUB 4240,4860,6020
4180  RETURN
4200  '
4220  ' *** Monoalphabetic Alphabet Entry Subroutine ***
4240  CLS:PLFAG=0:CIFLAG=0:DONEFLAG=0
4260  PRINT TAB(5);“Present alphabet is--”:PRINT
4280  PRINT TAB(10);“P: ”;
     :FOR N=1 TO 26
        :PRINT MID$(PCOMP$,N,1);“   ”;
     :NEXT N
4300  PRINT TAB(10);“C:   ”;
     :FOR N=1 TO 26
       :PRINT MID$(CCOMPO$,N,1);“   ”;
     :NEXT N
4320  PRINT
     :PRINT
4340  PRINT TAB(20);“1.  Change plain component”
4360  PRINT TAB(20);“2.  Change cipher component”
4380  PRINT TAB(20);“3.  Change specific key”
4400  PRINT TAB(20);“4.  Accept alphabet as shown”
4420  PRINT
     :PRINT TAB(18);“Enter your choice:   ”;
4440  INPUT CHOICE
4460  ON CHOICE GOSUB 4520,4580,4640,4500
4480  IF DONEFLAG=1
        THEN CCOMP$(1)=CCOMPO$
       :RETURN
      ELSE GOTO 4240 ' Exit if done
4500  DONEFLAG=1
     :RETURN
4520  ROW=3
     :COLUMN=11
     :PLFAG=1
     :GOSUB 5640
4540  PCOMP$=COMP$
4560  RETURN
4580  ROW=4
     :COLUMN=11
     :CIFLAG=1
     :GOSUB 5640
4600  CCOMPO$=COMP$

4620  RETURN
4640  LOCATE 11,10:X=SCREEN (3,13):
      PRINT “Type the specific key: ”;CHR$(X-32);
      “      of plaintext = ? of ciphertext.”
4660  LOCATE 11,50,1
```

```
4680  X$=INKEY$
     :IF X$=″ ″
        THEN 4680
4700  IF ASC(X$)>96 AND ASC(X$)<123
        THEN X$=CHR$(ASC(X$)-32)
4720  FOR N=1 TO 26:
        IF X$=MID$(CCOMPO$,N,1)
          THEN 4780
4740  NEXT N
4760  PRINT “CHARACTER NOT FOUND IN CIPHER COMPONENT”
     :GOTO 4640
4780  TCOMP$=RIGHT$(CCOMPO$,27-N)+LEFT$(CCOMPO$,N-1)
     :CCOMPO$=TCOMP$
4800  RETURN
4820  ′
4840  ′     *** Periodic and Aperiodic Alphabet Entry Subroutine ***
4860  CLS
     :DONEFLAG=0
     :PLFLAG=0
     :CIFLAG=0
4880  PRINT TAB(5);“Plain component is--”
4900  PRINT TAB(10);“P:   ”;
     :FOR N=1 TO 26
       :PRINT MID$(PCOMP$,N,1);“   ”;
     :NEXT N
     :PRINT
4920  PRINT TAB(5);“Cipher component is--”
4940  PRINT TAB(10);“C:   ”;
     :FOR N=1 TO 26
       :PRINT MID$(CCOMPO$,N,1);“   ”;
     :NEXT N
     :PRINT
     :PRINT
4960  IF AFLAG=0
        THEN PRINT TAB(5);“Length of period is:   ”;PERIOD
      ELSE PRINT TAB(5);“Length of key is:   ”;PERIOD
4980  X=SCREEN(2,13)
5000  IF AFLAG=0
        THEN REPEATKEY$=LEFT$(RKEY$,PERIOD)
5020  IF AFLAG=0
        THEN PRINT TAB(5);“Repeating key is   ”;CHR$(X-32);” of
        plaintext = ”;REPEATKEY$
       :PRINT
     :ELSE PRINT TAB (5);“Long running key is:   ”;REPEATKEY$
     :PRINT
5040  PRINT
     :PRINT
5060  PRINT TAB(20);“1. Change plain component”

5080  PRINT TAB(20);“2. Change cipher component”
```

```
5100  IF AFLAG=0
         THEN PRINT TAB (20);"3.  Change repeating key"
      ELSE PRINT TAB(20);"3.  Generate long running key"
5120  IF AFLAG=0
         THEN PRINT TAB(20);"4.  Show complete matrix"
      ELSE PRINT TAB(20);"4.  Accept alphabets"
5140  PRINT
     :PRINT TAB(18);"Enter your choice:   ";
5160  INPUT CHOICE
5180  ON CHOICE GOSUB 5220,5260,5300,5420
5200  IF DONEFLAG=1
         THEN RETURN
      ELSE GOTO 4860
5220  ROW=2
     :COLUMN=11
     :PLFLAG=1
     :GOSUB 5640
5240  PCOMP$=COMP$
     :RETURN
5260  ROW=4
     :COLUMN=11
     :CIFLAG=1
     :CMIXFLAG=1
     :GOSUB 5640
5280  CCOMPO$=COMP$
     :RETURN
5300  IF AFLAG=1
         THEN 5820
      ELSE LOCATE 7,39
     :INPUT RKEY$
5320  PERIOD=LEN(RKEY$)
5340  FOR N=1 TO PERIOD:
         FOR P=1 TO 26
            :IF MID$(RKEY$,N,1)=MID$(CCOMPO$,P,1)
               THEN 5380
5360     NEXT P
5380     CCOMP$(N)=RIGHT$(CCOMPO$,27-P)+LEFT$(CCOMPO$,P-1)
     :NEXT N
5400  RETURN
5420  CLS
     :IF AFLAG=1
        THEN 4500
5440  PRINT TAB(9);"P:   ";
     :FOR N=1 TO 26
        :PRINT MID$(PCOMP$,N,1);"   ";
     :NEXT N
     :PRINT
     :PRINT TAB(13);"----------------------------------------------------"
5460  FOR P=1 TO PERIOD
```

```
5480     PRINT TAB(9);“C”;CHR$(48+P);“:  ”;
        :FOR N=1 TO 26
           :PRINT MID$(CCOMP$(P),N,1);“   ”;
        :NEXT N
        :PRINT

5500  NEXT P
5520  PRINT TAB(20);“1. Change matrix”
5540  PRINT TAB(20);“2. Accept matrix”
5560  INPUT“               Enter your choice:   ”;CHOICE
5580  ON CHOICE GOTO 4860,4500
5600  '
5620  ' *** Reads in Edited Plain or Cipher Component From Screen ***
5640  LOCATE ROW, COLUMN
     :INPUT DUMMY$   ' DUMMY$ is not used as text is read from screen
5660  COMP$=" "
5680  FOR N=13 TO 63 STEP 2
        :X=SCREEN(ROW,N)
        :COMP$=COMP$+CHR$(X)
5700     IF PLFLAG=1 AND (X<96 OR X>122) AND X<>46
            THEN BEEP
           :GOTO 5640
5720     IF CIFLAG=1 AND (X<65 OR X>90)
            THEN BEEP
       :GOTO 5640
5740  NEXT N
5760  RETURN
5780  '
5800  ' *** Aperiodic Long-Running Key Generation Subroutine ***
5820  CLS
5840  RANDOMIZE
5860  INPUT “Enter the number of alphabets (up to 200):   ”;PERIOD
5880  FOR N=1 TO PERIOD
5900  LRK$=LRK$+CHR$(INT(RND*26)+65)
5920  NEXT N
5940  REPEATKEY$=LRK$
     :RKEY$=LRK$
5960  GOTO 5340
5980  '
6000  ' *** Sets Flag Indicating Long-Running Key System ***
6020  AFLAG=1
     :GOTO 4806
6040  '
6060  ' *** Text Print Subroutine ***
6080  CLS
6100  PRINT “IS PRINTER READY (Y/N)?   ”
6120  X$=INKEY$
     :IF X$=" "
        THEN 6120
```

```
6140  IF X$=“N” OR X$=“n”
        THEN RETURN
6160  OUTFILE$=PRINTER$
6180  GOSUB 6440
6200  PRINT #1,FORMFEED$;FORMFEED$
6220  CLOSE #1
6240  STATUS$(4)=“      (TEXT PRINTED)”
6260  IF PRINTER$<>“CON”
        THEN 6320
6280  PRINT “PRESS ANY KEY TO CONTINUE”
6300  GO$=INKEY$
     :IF GO$=" "
       THEN 6300
6320  RETURN
6340  '
6360  ' *** Text Save to Disk Subroutine ***
6380  CLS
6400  PRINT “Enter complete disk filename for the save text, for example,”
6420  INPUT “A:MYSAVE.TXT   ”;OUTFILE$
6440  OPEN OUTFILE$ FOR OUTPUT AS #1
6460  TEXTCOUNT=0
6480  FOR N=1 TO NRLINES
6500    PRINT #1,PTEXTD$(N)
6520    PRINT #1,CTEXTD$(N)
6540    PRINT #1,KTEXTD$(N)
6560    TEXTCOUNT=TEXTCOUNT+LEN(CTEXTI$(N))
6580    PRINT +1,
6600  NEXT N
6620  IF PERIOD>20
        THEN 6720
6640  PRINT#1,PCOMP$
6660  FOR N=1 TO PERIOD
6680    PRINT #1,CCOMP$(N)
6700  NEXT N
6720  IF OUTFILE$=PRINTER$ OR FILEFLAG=1 THEN RETURN
6740  CLOSE #1
6760  IF OUTFILE$<>PRINTER$ THEN STATUS$(5)=“      (TEXT SAVED)”
6780  RETURN
6800  '
6820  ' *** Frequency Count, IC Subroutine ***
6840  CLS
6860  PRINT “Select the routine you want to run:”
6880  PRINT:PRINT
6900  PRINT "      1. Monographic frequencies and ICs"+STAT$(1)
6920  PRINT "      2. Digraphic frequencies and ICs"+STAT$(2)
6940  PRINT "      3. Periodic frequencies and ICs"+STAT$(3)
6960  PRINT "      4. Chi test"+STAT$(4)
6980  PRINT "      5. RETURN TO MAIN MENU"
7000  INPUT "          Your choice: ",CHOICE$
```

```
7020  IF ASC (CHOICE$)<49 OR ASC(CHOICE$)>53
        THEN 7000
7040  ON (ASC(CHOICE$)-48) GOSUB 7120,7440,7900,11120, 1180
7060  GOTO 6840
7080  '

7100  ' *** Monographic Frequency and IC Subroutine ***
7120  FOR LINE=1 TO NRLINES
7140    FOR CHARPOS=1 TO LEN(CTEXTI$(LNE))
7160      NXTLTR$=MID$(CTEXTI$(LNE),CHARPOS,1)
7180      Z=ASC(NXTLTR$)-64
7200      MFREQ(Z)=MFREQ(Z)+1
7220    NEXT CHARPOS
7240  NEXT LNE
7260  FOR Z=1 TO 26
7280    TOTLTRS=TOTLTRS+MFREQ(Z)
7300    PHISUM=PHISUM+(MFREQ(Z)*(MFREQ(Z)-1))
7320  NEXT Z
7340  PHIMONO=26*PHISUM/(TOTLTRS*(TOTLTRS-1))
7360  MFLAG=1
     :STAT$(1)=" (COMPLETED)"
     :STATUS$(6)="      (COMPLETED)"
7380  RETURN
7400  '

7420  ' *** Digraphic Frequency and IC ***
7440  FOR LNE=1 TO NRLINES
7460    IF (LEN(CTEXTI$(LNE))/2-INT(LEN(CTEXTI$(LNE))/2))=0
          THEN 7520
7480    CARRY$=RIGHT$(CTEXTI$(LNE),1)
       :CTEXTI$(LNE)=LEFT$(CTEXTI$(LNE),LEN(CTEXTI$(LNE))-1)

7500    CTEXTI$(LNE+1)=CARRY$+CTEXTI$(LNE+1)
7520  NEXT LNE
7540  FOR LNE=1 TO NRLINES
7560    FOR DIG=1 TO INT(LEN(CTEXTI$(LNE))/2)
7580      LTR1=ASC(MID$(CTEXTI$(LNE),DIG*2-1,1))-64
         :LTR2=ASC(MID$(CTEXTI$(LNE),DIG*2,1))-64

7600      IF LTR1=-18 OR LTR2=-18
            THEN 7640
7620      DIFREQ(LTR1,LTR2)=DIFREQ(LTR1,LTR2)+1
7640    NEXT DIG
7660  NEXT LNE
7680  FOR ROW=1 TO 26
7700    FOR COLUMN=1 TO 26
7720      TOTDIG=TOTDIG+DIFREQ(ROW,COLUMN)
7740      DIPHISUM=DIPHISUM+(DIFREQ(ROW,COLUMN)*(DIFREQ(ROW,COLUMN)-1))
7760    NEXT COLUMN
7780  NEXT ROW
7800  PHIDIG=676*DIPHISUM/(TOTDIG*(TOTDIG-1))
```

```
7820  DFLAG=1:
     :STAT$(2)=" (COMPLETED)"
     :STATUS$(6)="  (COMPLETED)"
7840  RETURN
7860  '
7880  ' *** Periodic Frequency, IC Subroute ***
7900  CYCLEPOS=0
7920  INPUT “What period do you want to use? ”,PERIOD
7940  FOR N=1 TO PERIOD
7960    FOR M=1 TO 26
7980      PFREQ(N,M)=0
8000    NEXT M
8020    PERPHISUM(N)=0
       :PERTOTLTR(N)=0
8040  NEXT N
8060  FOR N=1 TO NRLINES
8080    FOR M=1 TO LEN(CTEXTI$(N))
8100      CYCLEPOS=CYCLEPOS+1
8120      IF CYCLEPOS>PERIOD
            THEN CYCLEPOS=1
8140      NXTCHAR$=MID$(CTEXTI$(N),M,1)
8160      Z=ASC(NXTCHAR$)-64
8180      IF Z=-18 THEN Z=27
8200      PFREQ(CYCLEPOS,Z)=PFREQ(CYCLEPOS,Z)+1
8220    NEXT M
8240  NEXT N
8260  FOR M=1 TO PERIOD
8280    FOR N=1 TO 26
8300      PERTOTLTR(M)=PERTOTLTR(M)+PFREQ(M,N)
8320      PERPHISUM(M)=PERPHISUM(M)+(PFREQ(M,N)*(PFREQ(M,N)-1))
8340    NEXT N
8360    PHIPERI(M)=26*PERPHISUM(M)/(PERTOTLTR(M)*(PERTOTLTR(M)-1))
8380  NEXT M
8400  PFLAG=1
     :STAT$(3)="  (COMPLETED)"
     :STATUS$(6)="  (COMPLETED)"
8420  IF CMIXFLAG=0
        THEN 8540 ' skips mixed alphabet routine if std sequence
8440  FOR M=1 TO PERIOD
8460    FOR N=1 TO 26
8480      PMIXFREQ(M,N)=PFREQ(M,ASC(MID$(CCOMPO$,N,1))-64)
8500    NEXT N
8520  NEXT M
8540  RETURN
8560  '
8580  ' *** Mixed Alphabet Periodic Stat Print ***
8600  ALPH$=" A B C D E F G H I J K L M N O P Q R S T U
      V W X Y Z"
8620  CLS
```

```
8640  OUTFILE$=PRINTER$
8660  GOSUB 6440
8680  IF MFLAG=1
         THEN GOSUB 8880
8700  IF DFLAG=1
         THEN PRINT #1,FORMFEED$
        :GOSUB 9080
8720  IF PFLAG=1
         THEN PRINT #1,FORMFEED$
        :GOSUB 9360
8740  IF CMIXFLAG=1
         THEN PRINT #1,FORMFEED$
        :GOSUB 9580
8760  PRINT #1,FORMFEED$
8780  PRINT #1,FORMFEED$
8800  CLOSE #1
8820  RETURN
8840  '
8860  ' *** Print Monographic Stats ***
8880  PRINT #1,
     :PRINT #1,
8900  PRINT #1,ALPH$
8920  FOR N=1 TO 26
8940     PRINT #1,USING "###";MFREQ(N);
8960  NEXT N
8980  PRINT #1,
     :PRINT #1,
9000  PRINT #1,"TOTAL LETTERS =";TOTLTRS;"   MONOGRAPHIC IC =";PHIMONO
9020  RETURN
9040  '
9060  ' ** Print Digraphic Stats **
9080  PRINT #1,
     :PRINT #1,
9100  PRINT #1, "  ";ALPH$
9120  FOR N=1 TO 26
9140     PRINT #1,CHR$(N+64);
9160     FOR M=1 TO 26
9180     PRINT #1,USING "###";DIFREQ(N,M);
9200     NEXT M
9220     PRINT #1,
9240  NEXT N
9260  PRINT #1,
     :PRINT #1,
9280  PRINT #1, "TOTAL DIGRAPHS =";TOTDIG;"   DIGRAPHIC IC=";PHIDIG
9300  RETURN
9320  '
9340  ' *** Print Monographic Stats ***
9360  PRINT #1,
     :PRINT #1,
9380  FOR N=1 TO PERIOD
9400    PRINT #1,ALPH$
9420    FOR M=1 TO 26
9440      PRINT #1,USING "###";PFREQ(N,M);
9460    NEXT M
9480    PRINT #1,
9500    PRINT #1,"TOTAL LETTERS =";PERTOTLTR(N);"        IC=";PHIPERI(N)
9520    PRINT #1,
       :PRINT #1,
9540  NEXT N
9560  RETURN
9580  PRINT#1,
     :PRINT #1,
9600  FOR M=1 TO PERIOD
9620    ALPHMIX$(M)=" "
9640    FOR N=1 TO 26
9660      ALPHMIX$(M)=ALPHMIX$(M)+"   "+MID$(CCOMPO$,N,1)
9680    NEXT N
9700  NEXT M
9720  FOR M=1 TO PERIOD
9740    PRINT #1,ALPHMIX$(M)
9760    FOR N=1 TO 26
9780      PRINT #1,USING "###";PMIXFREQ(M,N);
9800    NEXT N
9820    PRINT #1,
9840    PRINT #1, "TOTAL LETTERS =";PERTOTLTR(M);"        IC =";PHIPERI(M)
9860    PRINT #1,
       :PRINT #1,
9880  NEXT M
9900  RETURN
9920  '
9940  ' *** Statistics Save to Disk Subroutine ***
9960  ALPH$=" A B C D E F G H I J K L M O P Q R S T U
      V W X Y Z"
9980  CLS
10000  PRINT "Enter the complete disk filename for the saved statistics, for example,"
10020  INPUT "A:MYSTAT.TXT ";OUTFILE$
10040  FILEFLAG=1
10060  GOSUB 6440
10080  IF MFLAG=1
         THEN GOSUB 8880
10100  IF DFLAG=1
         THEN GOSUB 9080
10120  IF PFLAG=1
         THEN GOSUB 9360
10140  IF CMIXFLAG=1
         THEN GOSUB 9580
10160  CLOSE #1
10180  RETURN
```

```
10200  '
10220  ' *** Subroutine to Find Repeats ***
10240  INPUT “What is the shortest length repeat you want listed?”,RPTLEN
10260  OUTFILE$=PRINTER$
10280  OPEN OUTFILE$ FOR OUTPUT AS #1
10300  IF RPTLEN<2
         THEN 10240
10320  FOR TLINE=1 TO NRLINES-1
10340    FOR ALTR=1 TO LEN(CTEXTI$(TLINE))
10360      IF TLINE<>NRLINES
             THEN CT$=CTEXTI$(TLINE)+CTEXTI$(TLINE+1)
           ELSE CT$=CTEXTI$(TLINE)
10380      A$=MID$(CT$,ALTR,RPTLEN)
10400      FOR BLTR=ALTR+2 TO LEN(CTEXTI$(TLINE))+2
             :BLINE=TLINE
             :CTB$=CT$
10420        IF BLTR>LEN(CTEXTI$(TLINE))
               THEN 10480
10440        B$=MID$(CTB$,BLTR,RPTLEN)
10460        IF A$=B$
               THEN GOSUB 10800
10480      NEXT BLTR
10500      IF TLINE=NRLINES
             THEN 10660
10520      FOR BLINE=TLINE+1 TO NRLINES
10540        IF BLINE<>NRLINES
               THEN CTB$=CTEXTI$(BLINE)+CTEXTI$(BLINE+1)
             ELSE CTB$=CTEXTI$(BLINE)
10560        FOR BLTR=1 TO LEN(CTEXTI$(BLINE))
10580          B$=MID$(CTB$,BLTR,RPTLEN)
10600          IF A$=B$
                 THEN GOSUB 10800
10620        NEXT BLTR
10640      NEXT BLINE
10660    NEXT ALTR
10680  NEXT TLINE
10700  PRINT #1, FORMFEED$,FORMFEED$
10720  CLOSE #1
10740  RETURN
10760  '
10780  ' *** Subroutine to Check Length of Repeat and Print It ***
10800  LONGER=RPTLEN
10820  PRINT A$
10840  LONGER=LONGER+1
10860  IF MID$(CT$,ALTR,LONGER)=MID$(CTB$,BLTR,LONGER)
         THEN 10840 ' Try it longer
10880  LONGER=LONGER-1 ' Nope, too long
10900  PRINT #1,MID$(CT$,ALTR,LONGER);" AT LINE";TLINE;", LETTER";ALTR;
       " AND AT LINE";BLINE;", LETTER";BLTR
```

```
10920  RETURN
10940  '
10960  ' *** Quit Subroutine ***
10980  CLS
11000  INPUT “Are you sure you want to quit (Y/N)? ”,CHOICE$
11020  IF CHOICE$ <>"Y" AND CHOICE$ <> "y"
         THEN 1180
11040  KEY ON ' restores bottom of screen prompts
11060  END
11080  '
11100  ' *** Chi Test Subroutine ***
11120  PRINT “Do you want to print results or save to disk as text file?”
11140  INPUT “Enter P for printer, D for disk, or Q to quit.”,S$
11160  IF S$=“P” OR S$=“p”
         THEN OUTFILE$=PRINTER$
        :GOTO 11240
11180  IF S$=“Q” OR S$=“q”
         THEN RETURN
11200  IF S$<>“D” AND S$<>“d”
         THEN 11140
11220  INPUT “Enter the complete disk filename. ”,OUTFILE$
11240  OPEN OUTFILE$ FOR OUTPUT AS #1
11260  PRINT “Which of the ”;PERIOD;“alphabets do you want to match?”
11280  PRINT
11300  INPUT “     Enter number of 1st alphabet to be matched: ”,ALF1
11320  INPUT “     Enter number of 2nd alphabet to be matched: ”,ALF2
11340  PRINT “MATCHING ALPHABET”;ALF1;“AND ALPHABET”;ALF2
11360  PRINT #1,“MATCHING ALPHABET”;ALF1;“AND ALPHABET”;ALF2
11380  FOR N=1 TO 26
11400    IF CMIXFLAG=1
           THEN SET1(N)=PMIXFREQ(ALF1,N)
         ELSE SET1(N)=PFREQ(ALF1,N)
11420    IF CMIXFLAG=1
           THEN SET2(N)=PMIXFREQ(ALF2,N)
         ELSE SET2(N)=PFREQ(ALF2,N)
11440  NEXT N
11460  FOR M=1 TO 26
11480    FOR L=1 TO 26
11500      PRINT #1,"   "MID$(CCOMPO$,L,1);  ' Print first sequence
11520    NEXT L
11540    PRINT #1,
11560    FOR L=1 TO 26
11580      PRINT #1, USING "###";SET1(L); ' Print first sequence frequencies
11600    NEXT L
11620    PRINT #1,
11640    FOR L=0 TO 25
11660      LTRPOS=M+L
          :IF LTRPOS>26
              THEN LTRPOS=LTRPOS-26
```

```
11680      PRINT #1, "   ";MID$(CCOMPO$,LTRPOS,1); ' Print second sequence
11700    NEXT L
11720    PRINT #1,
11740    MATCH(M)=0
11760    FOR N=1 TO 26
11780      MATCH(M)=MATCH(M)+(SET1(N)*SET(N))
11800      PRINT #1, USING "###";SET2(N); ' Print second sequence frequencies
11820    NEXT N
11840    PRINT #1,
11860    IF M/2-INT(M/2)<>0
           THEN PRINT TAB(1) "MATCH";M;":";MATCH (M);
         ELSE PRINT TAB(40) "MATCH";M;":";MATCH (M);
11880    PRINT #1,"          MATCH";M;":";MATCH (M)
        :PRINT #1,
11900    SET2(27)=SET2(1)
11920    FOR N=1 TO 26
11940      SET2(N)=SET2(N+1):
         NEXT N
11960  NEXT M
11980  IF OUTFILE$=PRINTER$
         THEN PRINT #1,FORMFEED$
12000  INPUT "ANOTHER MATCH (Y/N)?",Q$
12020  IF Q$="Y" OR Q$="y"
         THEN 11300
12040  IF OUTFILE$=PRINTER$
         THEN PRINT #1,FORMFEED$
12060  CLOSE #1
12080  RETURN
```