24 FEB 1969

AD 683474

COPY NO. 110

TECHNICAL MEMORANDUM 1871

EDITOR

A COMPUTER PROGRAM FOR DOCUMENTATION PURPOSES

WILLIAM H. BOLTE

MAR 7 1969

DECEMBER 1968

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Best Available Copy

PICATINNY ARSENAL
DOVER, NEW JERSEY

Reproduced by the
CLEARINGHOUSE
for Federal Scientific & Technical

TECHNICAL MEMORANDUM 1871

EDITOR - A COMPUTER PROGRAM FOR DOCUMENTATION PURPOSES

BY

WILLIAM H. ECLTE

DECEMBER 1968

DATA PROCESSING SYSTEMS OFFICE
PICATINNY ARSENAL
DOVER, N.J.

TABLE OF CONTENTS

SECTION	PAGE
1. ABSTRACT	1
2. INTRODUCTION	2
3. GENERAL DESCRIPTION	3
4. CODE DESCRIPTION & APPLICATION	4
A. EDITOR PROGRAM CODES	4
B. INDIVIDUAL SAMPLE CODE ILLUSTRATIONS	7
C. ILLUSTRATIVE EXAMPLES	11
5. APPENDICES	15
A. FLOW CHART OF BASIC PROGRAM LOGIC	16
B. LISTING OF FORTRAN IV SOURCE PROGRAM	21
6. TABLE OF DISTRIBUTION	26

1. ABSTRACT

THIS REPORT DESCRIBES THE APPLICATION OF AN EDITING PROGRAM CALLED 'EDITOR' FOR DOCUMENTATION PURPOSES. THIS PROGRAM WAS WRITTEN IN FORTRAN IV LANGUAGE FOR THE IBM-360/65 DIGITAL COMPUTER BY MR. SIDNEY KRAVITZ (MECHANICAL ENGINEER) OF THE DATA PROCESSING SYSTEMS OFFICE, PICATINNY ARSENAL. IT REQUIRES A TOTAL MEMORY LENGTH OF 20,582 BYTES. THIS REPORT ENJOYS THE UNIQUE DISTINCTION OF HAVING BEEN PHYSICALLY PREPARED BY THE VERY PROCESS IT DESCRIBES.

THIS PROGRAM IS PRESENTLY BEING USED ON AN EXPERIMENTAL BASIS TO EVALUATE ITS FEASIBILITY FOR ARSENAL WIDE DISTRIBUTION.

2. INTRODUCTION

PROGRAM DOCUMENTATION CAN BE AND OFTEN IS A TEDIOUS, ALBEIT NECESSARY, TASK UNDERTAKEN AT THE COMPLETION OF DEVELOPING A COMPUTER PROGRAM. 'EDITOR' WAS WRITTEN TO EXPEDITE THIS TASK.

'EDITOR' OFFERS A SOLUTION TO THE SHORTAGE OF AVAILABLE TYPISTS AND SECRETARIES. IT TAKES ADVANTAGE OF THE IBM-1403 HIGH SPEED PRINTER LOCATED AT THE PICATINNY COMPUTER CENTER, THUS ENABLING THE USER TO HAVE HIS REPORTS 'TYPED' BY THE COMPUTER. ALL TEXT EDITING INSTRUCTIONS (LINE SPACING, UNDERLINING, CENTERING, PAGE NUMBERING, ETC...) ARE PASSED TO THE 'EDITOR' PROGRAM BY MEANS OF ALPHA NUMERICAL CODES PUNCHED IN COLUMN 1 OF THE INPUT DATA CARDS. THE DATA CARDS CONTAIN THE TEXT OF THE REPORT, HAVING BEEN PUNCHED FROM FORTRAN CODING SHEETS UPON WHICH THE AUTHOR HAS 'WRITTEN' HIS REPORT.

THIS PROGRAM IS RATHER SIMPLE AND EASY TO USE. HOWEVER, IT IS FELT THAT BETTER UNDERSTANDING OF THE INDIVIDUAL PROGRAM CODES WILL BE ACHIEVED BY STUDYING ACTUAL SAMPLES AND VARIATIONS OF EACH. THEREFORE, THE LATTER PART OF THIS REPORT WILL CONTAIN SAMPLES OF MOST OF THE PROGRAM CODES AND THEIR EFFECTS AS WELL AS AN ILLUSTRATIVE EXAMPLE OF PART OF THE ACTUAL CODED DECK USED FOR THE DOCUMENTATION OF THIS REPORT.

3. GENERAL DESCRIPTION

THIS PROGRAM WAS DESIGNED TO ALLOW THE PRINTED OUTPUT PAGES TO FALL WITHIN THE COMMON GOVERNMENT STANDARD (8 X 10-1/2 INCHES) PAGE SIZE. PAGE NUMBERING IS ACCOMPLISHED AUTOMATICALLY STARTING AFTER THE FIRST PAGE, SINCE THIS IS OFTEN THE TITLE PAGE. ROMAN NUMERAL NUMBERING MAY BE INITATED AT THE OPTION OF THE USER UP TO A MAXIMUM OF XVII. ARABIC NUMERAL NUMBERING THEN BEGINS AND CONTINUES UNTIL THE END OF THIS REPORT. IF DESIRED, ADDITIONAL REPORTS CAN IMMEDIATELY FOLLOW, SEPARATED BY THE APPROPIATE LOGIC CONTROL CARDS.

NORMALLY 24 LINES OF PRINTED INFORMATION ARE PRODUCED PER PAGE OF OUTPUT (DOUBLE SPACING). IF PAGE OVERFLOW OCCURS, THE INFORMATION IS CONTINUED ON THE NEXT PAGE WITH THE PROPER PAGE NUMBER SEQUENCE.

THIS PROGRAM PRODUCES OUTPUT ON A ONE TO ONE BASIS, THAT IS, ONE PUNCHED CARD OF INFORMATION PRODUCES ONE LINE OF OUTPUT INFORMATION. ALL PUNCHED CARDS DO NOT NECESSARILY REQUIRE PROGRAM CODES (IN COLUMN 1) ALL THE TIME. ONLY WHEN A CERTAIN OPERATION CATAGORIZED BY ONE OF THE PROGRAM CODES IS DESIRED, SHOULD IT APPEAR, BEING SUPPLIED BY THE USER.

4. CODE DESCRIPTION & APPLICATION

4.A. EDITOR PROGRAM CODES

CODES	MEANING
A	CENTER & UNDERLINE THE LINE
C	CENTER THE LINE
F	LAST LINE ON THE LAST PAGE OF THE REPORT (A NEW REPORT MAY BE PLACED IMMEDIATELY AFTER THIS CARD)
L	LAST LINE ON A PAGE
S	SELECTIVE UNDERLINING - UNDERSCORES ONLY (BUT ALL OF) THE CHARACTERS FOUND ON THE CARD.
U	UNDERLINES EVERYTHING BETWEEN THE LEFTMOST CHARACTER AND THE RIGHTMOST CHARACTER OF THE LINE
V	THE FIRST 'V' PLACES THE PROGRAM IN THE 'V-MODE'. THE SECOND 'V' REMOVES IT FROM THE 'V-MODE'. THE THIRD 'V' PLACES THE PROGRAM BACK INTO THE 'V-MODE' AGAIN, ETC.. WHEN IN THE V-MODE, THE PROGRAM WILL LIST ALL 80 COLUMNS OF CHARACTERS AND SINGLE SPACE ALL LINES. ALL CODES FOUND IN COLUMN 1 ARE DISREGARDED AS COMMANDS BUT ARE NEVERTHELESS LISTED. THE ONLY EXCEPTION TO THE RULE IS THE 'V'-CODE ITSELF. A 'V' WILL NEVER BE LISTED IN COLUMN 1 OF THE LISTING WHETHER IN THE 'V-MODE' OR NOT. HOWEVER, PAGE NUMBERING STILL CONTINUES.

CODES	MEANING
DIGIT	ANY ODD DIGIT BETWEEN 1 AND 9 WILL CAUSE THE PROGRAM TO SKIP (DIGIT+1) LINES BEFORE PRINTING THE INFORMATION FOUND ON THAT CARD.
DIGIT	ANY EVEN DIGIT BETWEEN 2 AND 10 WILL CAUSE THE PROGRAM TO SKIP (DIGIT+1) LINES AFTER PRINTING THE INFORMATION FOUND ON THAT CARD.
N	ALLOWS SINGLE SPACING (EVEN THOUGH DOUBLE SPACING NORMALLY OCCURS - IT IS OVERRIDEN IN THIS CASE).
R	ROMAN NUMERAL NUMBERING IS NORMALLY (AUTOMATICALLY) INTIATED BY THE PROGRAM STARTING WITH THE SECOND PAGE OF THE REPORT. 'R' DENOTES THE LAST LINE OF THE LAST PAGE FOR ROMAN NUMERALS. IF NO 'R' APPEARS, THE PROGRAM WILL BEGIN PAGE 1 AFTER PAGE XVII. A SECOND 'R' IN THE SAME REPORT WILL BE DISREGARDED BY THE PROGRAM. WHEN THE INPUT DATA IS PRECEDED BY AN 'R' CODE, PROGRAM CONTROL SKIPS ROMAN NUMERAL NUMBERING AND EMPLOYS NUMERICAL NUMBERING ONLY, AGAIN STARTING WITH THE SECOND PAGE OF THE REPORT.

THE FOLLOWING ARE SOME POINTERS TO REMEMBER CONCERNING THE USE OF 'EDITOR'.

1) NOT ALL CARDS REQUIRE A PROGRAM CODE.

2) REQUIRED CODES ARE TO BE PUNCHED IN COLUMN 1 ONLY.

3) A 'V' IN COLUMN 1 IS NEVER PRINTED ON THE OUTPUT LISTING REGARDLESS OF WHETHER THE PROGRAM IS IN THE 'V-MODE' OR NOT.

4) ONE PUNCHED CARD PRODUCES ONE LINE OF OUTPUT.

5) ROMAN NUMERAL NUMBERING CANNOT EXCEED XVII.

6) WITH PROPER USE OF THE 'DIGIT' & 'N' CODES ANY NUMBER OF LINES CAN FOLLOW OR PRECEDE A LINE OF OUTPUT.

7) IF THE USER PUNCHES LINE INFORMATION BETWEEN COLUMNS 7 TO 72 INCLUSIVELY, PROPER PAGE ALIGNMENT WILL OCCUR. WRITING BEYOND COLUMN 72 COULD NECESSITATE THE NEED FOR PHOTOGRAPHIC REDUCTION IN ORDER FOR IT TO FALL WITHIN THE 8 X 10-1/2 INCH LIMITS.

8) UNLESS SPECIFIED BY A PROGRAM CODE, DOUBLE SPACING NORMALLY OCCURS.

4.8. INDIVIDUAL SAMPLE CODE ILLUSTRATIONS

1. CODE - "A" - CENTER & UNDERLINE

CARD:

```
A     CENTER & UNDERLINE EXAMPLE
```

RESULT:

CENTER & UNDERLINE EXAMPLE

2. CODE - "C" - CENTERS THE LINE

CARD:

```
C     CENTER EXAMPLE
```

RESULT:

CENTER EXAMPLE

3. CODE - "S" - SELECTIVE UNDERLINING

CARD:

```
S     SELECTIVE UNDERLINING EXAMPLE
```

RESULT:

SELECTIVE UNDERLINING EXAMPLE

4. CODE - "U" - UNDERLINE

CARD:

```
U     UNDERLINE EXAMPLE
```

RESULT:

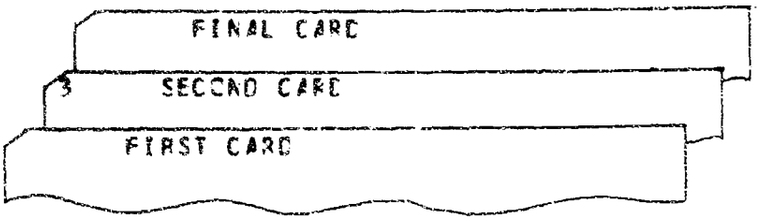
UNDERLINE EXAMPLE

4.B. INDIV. SAMPLE CODE ILLUS. (CONT.)

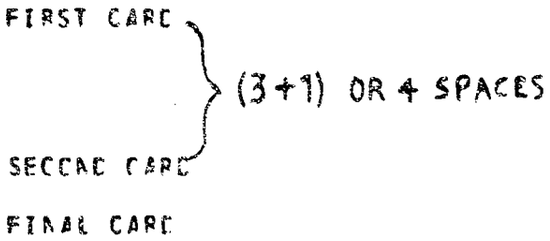
5. CODE - "DIGIT" - MULTIPLE SPACING

A) ODD NUMBER

CARDS:

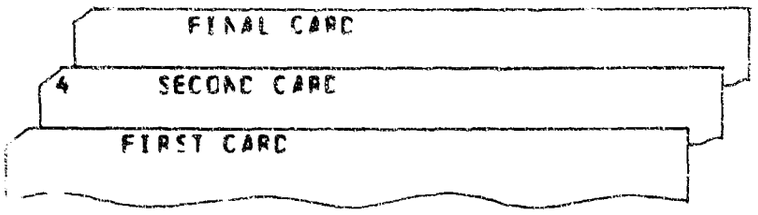


RESULT:

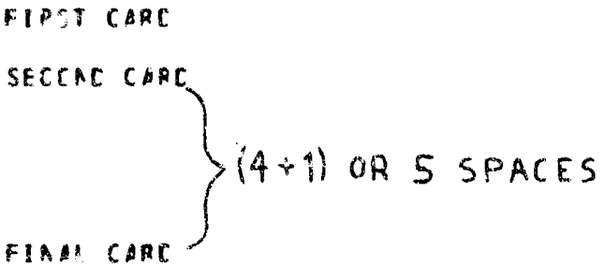


B) EVEN NUMBER

CARDS:



RESULT:



6. CODE - "V" - VERBATUM MODE

CARDS:

```
V
S     END 'V' MODE
4     817CCLC //*
A     2LZ 3C8FFL TXY
V     BEGIN 'V' MODE
```

RESULT:

```
                    BEGIN 'V' MODE
A                   2LZ 3C8FFL TXY
4                   817CCLC //*
S                   END 'V' MODE
```

7. CODE - "N" - NO SKIP - SINGLE SPACE

CARDS:

```
      NO SKIP EXAMPLE
N     SINGLE SPACE
```

RESULT:

```
SINGLE SPACE
NO SKIP EXAMPLE
```

8. CODE - "F" - LAST LINE OF LAST PAGE OF REPORT

CARD:

```
F     (LINE DATA PRINTED IF PRESENT)
```

RESULT:

REINITIALIZES PAGE NUMBERING SEQUENCE

9. CODE - "R" - LAST PAGE OF ROMAN NUMERALS

CARD:

```
R     (LINE DATA PRINTED IF PRESENT)
```

RESULT:

THE PAGE FOLLOWING BEGINS PAGE NUMBERING WITH 1

10. CODE - "L" - AN EXAMPLE OF THE 'L' CODE (LAST LINE ON PAGE) CAN BE FOUND IN SECTION 4.C.

4.C. ILLUSTRATIVE EXAMPLES

1. THE FOLLOWING IS THE LISTING OF THE TABLE OF CONTENTS OF THE ACTUAL PROGRAM CODED DECK USED FOR THE DOCUMENTATION OF THIS REPORT. IT WAS LISTED UNDER THE 'V-MODE'.

TABLE OF CONTENTS

SECTION	PAGE
1. ABSTRACT	1
2. INTRODUCTION	2
3. GENERAL DESCRIPTION	3
4. CODE DESCRIPTION & APPLICATION	5
A. EDITOR PROGRAM CODES	5
B. INDIVIDUAL SAMPLE CODE ILLUSTRATIONS	7
C. ILLUSTRATIVE EXAMPLES	11
5. APPENDICES	15
A. FLOW CHART OF BASIC PROGRAM LOGIC	16
B. LISTING OF FORTRAN IV SOURCE PROGRAM	21
6. TABLE OF DISTRIBUTION	26

2. THE FOLLOWING IS THE LISTING OF SECTION 4.B. OF THE ACTUAL PROGRAM CODED DECK USED FOR THE DOCUMENTATION OF THIS REPORT. IT WAS LISTED UNDER THE 'V-MODE'.

```
U       4.B. INDIVIDUAL SAMPLE CODE ILUSTRATIONS
N
          1. CODE - **A** - CENTER & UNDERLINE
N
N              CARD:
                          A     CENTER & UNDERLINE EXAMPLE
N
N            RESULT:
A       CENTER & UNDERLINE EXAMPLE
N
N
          2. CODE - **C** - CENTERS THE LINE
N
N              CARD:
                          C     CENTER EXAMPLE
N
N            RESULT:
C       CENTER EXAMPLE
N
N
          3. CODE - **S** - SELECTIVE UNDERLINING
N
N              CARD:
                          S     SELECTIVE UNDERLINING EXAMPLE
N            RESULT:
S                         SELECTIVE  UNDERLINING EXAMPLE
N
N
          4. CODE - **U** - UNDERLINE
N
N              CARD:
                          U     UNDERLINE EXAMPLE
```

4.C. ILLUS. EXAMPLES (CONT.)

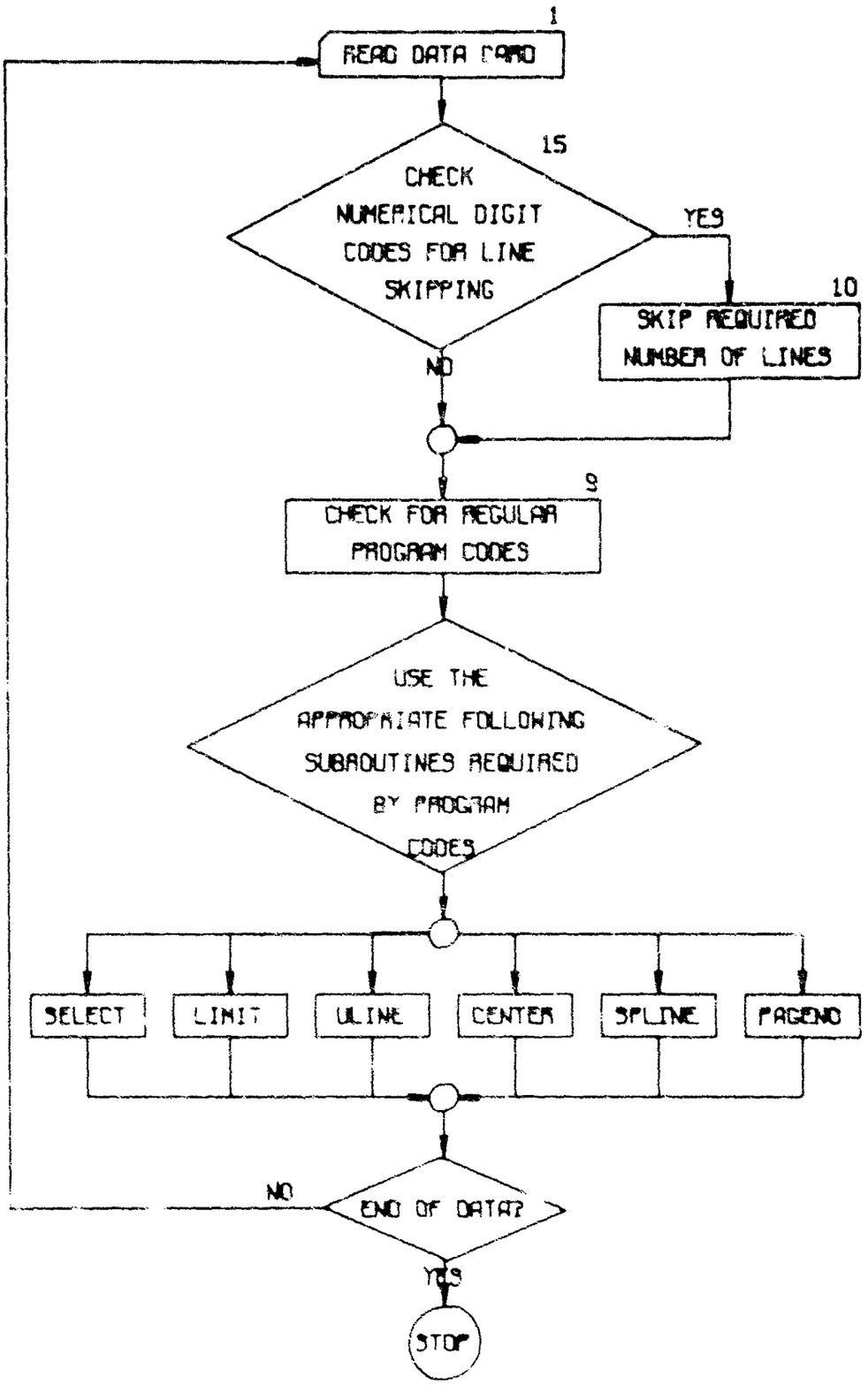
```
N          RESULT:
U                                UNDERLINE EXAMPLE
L
U      4.B. INDIV. SAMPLE CODE ILLUS. (CONT.)
N
         5. CODE - ''DIGIT'' - MULTIPLE SPACING
           A) ODD NUMBER
N            CARDS:
                                  FINAL CARD
                          3     SECOND CARD
                              FIRST CARD
N
N            RESULT:
                                    FIRST CARD
3                                   SECOND CARD
                                    FINAL CARD
N
           B) EVEN NUMBER
N            CARDS:
                                  FINAL CARD
                          4     SECOND CARD
                              FIRST CARD
N
N            RESULT:
                                    FIRST CARD
4                                   SECOND CARD
                                    FINAL CARD
L
U      4.B. INDIV. SAMPLE CODE ILLUS. (CONT.)
N
         6. CODE - ''V'' - VERBATOM MODE
N            CARDS:
                                  V
                                S     END 'V' MODE
                              4     B17COLD //*
                            A     2LZ 3DBFPL TXY
                          V     BEGIN 'V' MODE
N
N            RESULT:
```

```
                                BEGIN 'V' MODE
A                               2LZ 3D8FPL TXY
4                               817COLD //*
S                               END 'V' MODE

N
N
N
```

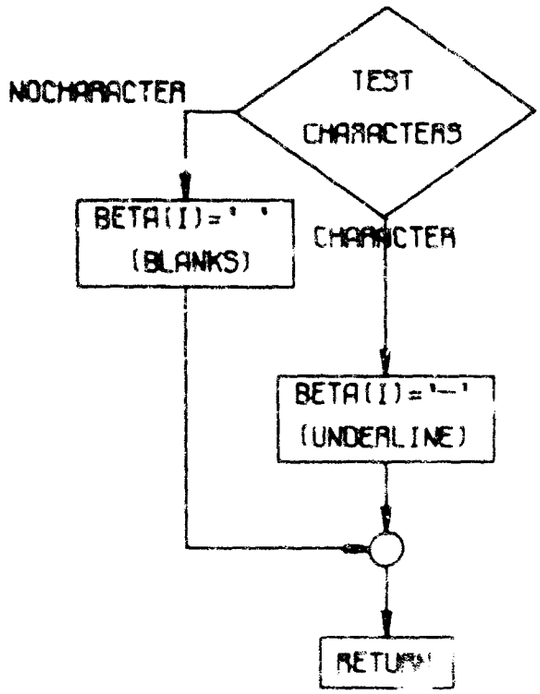
5. APPENDICES

5.A. FLOW CHART OF BASIC PROGRAM LOGIC

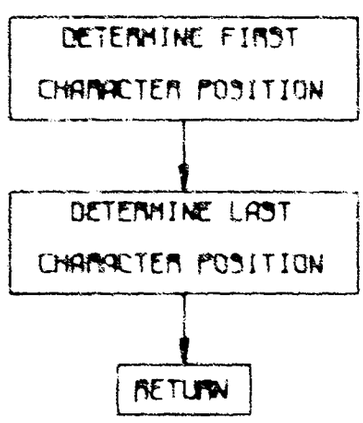


FLOW CHART OF BASIC LOGIC

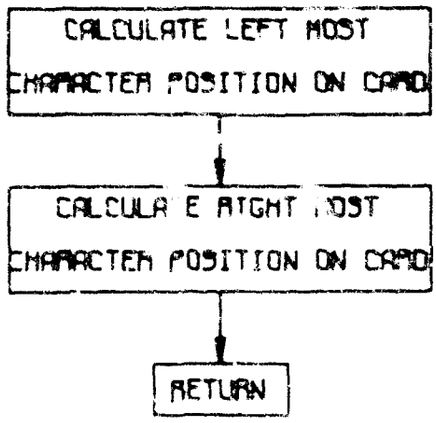
SUBROUTINE SELECT



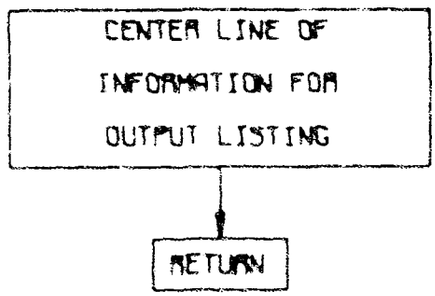
SUBROUTINE LIMIT



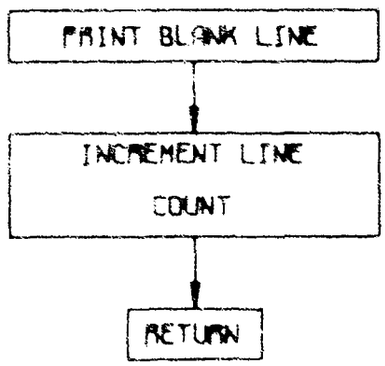
SUBROUTINE ULINE



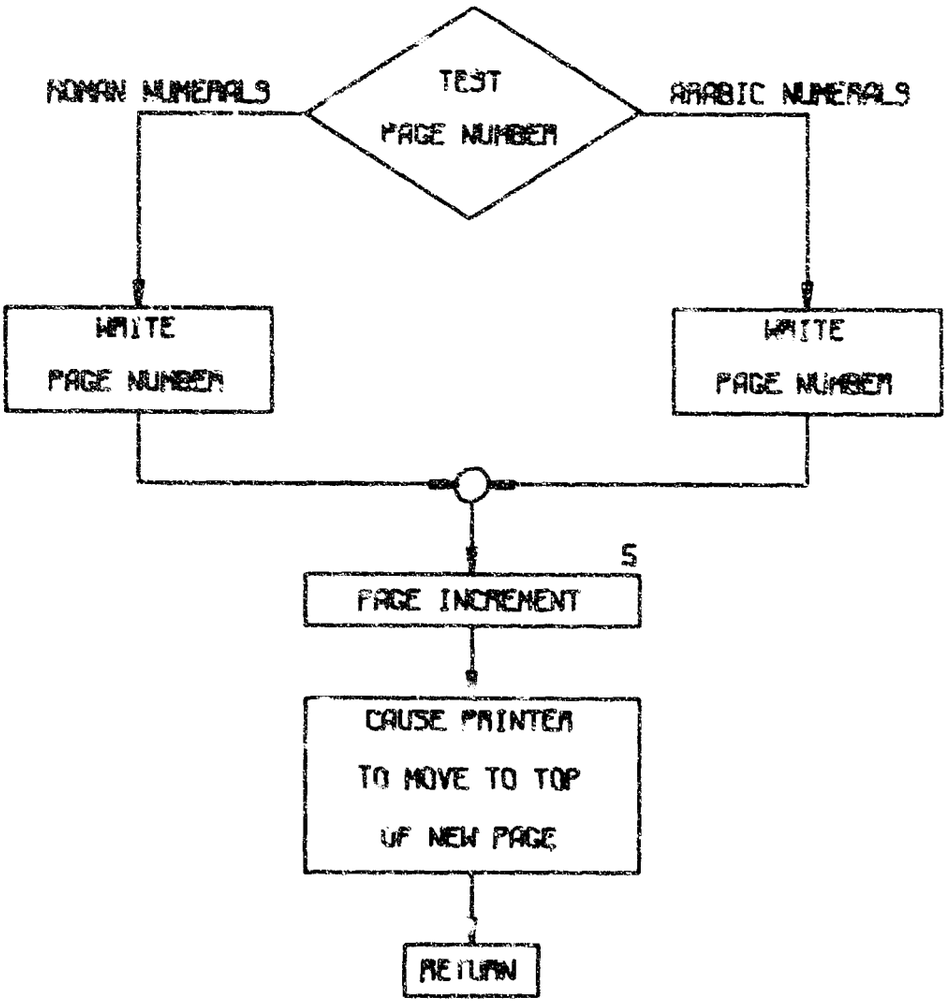
SUBROUTINE CENTER



SUBROUTINE SPLINE



SUBROUTINE PAGENO



5.8. LISTING OF FORTRAN IV SOURCE PROGRAM

```
C  EDITOR PROGRAM, SECOND VERSION, SEPTEMBER 1980
      DIMENSION ALPHA(80),BETA(80),ROMAN(20)
      DATA C/'C'/, U/'U'/, A/'A'/, XN/'N'/, R/'R'/, XL/'L'/, P/'P'/,
     1 PAGE/'PAGE'/, X1/'1'/, X2/'2'/, X3/'3'/, X4/'4'/, X5/'5'/,
     2 X6/'6'/, X7/'7'/, X8/'8'/, X9/'9'/,V/'V'/,SPACE/' '/,F/'F'/
     3,S/'S'/
   20 LMAN=0
      IA=0
      LINE=1
      LROMAN=0
      WRITE(6,8)
    8 FORMAT('1'////)
    1 READ(5,2,END=50)XC,(ALPHA(I),I=1,79)
    2 FORMAT(    80A1)
    3 FORMAT(17X,79A1,/)
    4 FORMAT(17X,79A1)
      IF(XC.EQ.V) IA=IA+1
      IW=IA/2
      IX=IA-2*IW
      IF(IX.NE.1) GO TO 15
      IF(XC.EQ.V)XC=SPACE
      BETA(1)=XC
      DO 22 I=1,79
      BETA(I+1)=ALPHA(I)
   22 CONTINUE
      WRITE(6,23) (BETA(I),I=1,80)
   23 FORMAT(17X,80A1)
      LINE=LINE+1
      GO TO 17
   15 LSKIP=0
      IF(XC.EQ.X1) LSKIP=1
      IF(XC.EQ.X2) LSKIP=2
      IF(XC.EQ.X3) LSKIP=3
      IF(XC.EQ.X4) LSKIP=4
      IF(XC.EQ.X5) LSKIP=5
      IF(XC.EQ.X6) LSKIP=6
      IF(XC.EQ.X7) LSKIP=7
      IF(XC.EQ.X8) LSKIP=8
      IF(XC.EQ.X9) LSKIP=9
      LSKIQ=LSKIP/2
      LSKIR=LSKIP-2*LSKIQ
      IF(LSKIR)9,9,10
   10 DO 11 I=1,LSKIP
      CALL SPLINE(LINE)
      IF(LINE.EQ.50) CALL PAGEND(LROMAN,LINE)
```

```
11 CONTINUE
 9 IF(XC.EQ.C)CALL LIMIT(ALPHA,NR,NL)
   IF(XC.EQ.U)CALL LIMIT(ALPHA,NR,NL)
   IF(XC.EQ.A)CALL LIMIT(ALPHA,NR,NL)
   IF(XC.EQ.C)CALL CENTER(ALPHA,NR,NL)
   IF(XC.EQ.A)CALL CENTER(ALPHA,NR,NL)
   IF(XC.EQ.A)CALL LIMIT(ALPHA,NR,NL)
   IF(XC.EQ.U)CALL ULINE(BETA,NR,NL)
   IF(XC.EQ.A)CALL ULINE(BETA,NR,NL)
   IF(XC.EQ.U) GO TO 6
   IF(XC.EQ.A) GO TO 6
   IF(XC.EQ.XN) GO TO 6
   IF(XC.EQ.S) CALL SELECT(ALPHA,BETA)
   IF(XC.EQ.S) GO TO 6
 5 WRITE(6,3)(ALPHA(I),I=1,79)
   LINE=LINE+2
   GO TO 12
 6 WRITE(6,4)(ALPHA(I),I=1,79)
   LINE=LINE+1
   IF(XC.EQ.U)WRITE(6,7)(BETA(I),I=1,79)
   IF(XC.EQ.A)WRITE(6,7)(BE A(I),I=1,79)
   IF(XC.EQ.S) WRITE(6,7) (BETA(I), I=1,79)
   IF(XC.EQ.U) LINE=LINE+1
   IF(XC.EQ.A) LINE=LINE+1
   IF(XC.EQ.S)  LINE=LINE+1
 7 FORMAT(1H+,16X,79A1,/)
12 IF(LSKIP.EQ.0) GO TO 13
   IF(LSKIR)13,21,13
21 DO 14 I=1,LSKIP
   CALL SPLINE(LINE)
   IF(LINE.EQ.50) CALL PAGENO(LROMAN,LINE)
14 CONTINUE
13 IF(XC.EQ.R) GO TO 16
   IF(XC.EQ.F) GO TO 16
   IF(XC.EQ.XL)GO TO 16
17 IF(LINE.EQ.49) CALL SPLINE(LINE)
   IF(LINE.EQ.50) CALL PAGENO(LROMAN,LINE)
   GO TO 19
16 IF(LINE.EQ.50) GO TO 17
     'NE=LINE+1
   DO 18 I=  INE,49
   CALL SPLINE(LINE)
18 CONTINUE
   GO TO 17
19 IF(XC.EQ.F) GO TO 20
   IF(LMAN.EQ.1) GO TO 1
   IF(XC.EQ.R) LMAN=1
   IF(XC.EQ.R) LROMAN=18
```

```
      GO TO 1
   50 STOP
      END



      SUBROUTINE SELECT(ALPHA,BETA)
      DIMENSION ALPHA(80),BETA(80)
      DATA RANK/' '/,HANK/'_'/
      DO 1 I=1,79
      IF(ALPHA(I).EQ.RANK) BETA(I)=RANK
      IF(ALPHA(I).NE.RANK) BETA(I)=HANK
    1 CONTINUE
      RETURN
      END



      SUBROUTINE LIMIT(ALPHA,NR,NL)
      DIMENSION ALPHA(80)
      DATA BLANK/' '/
      NL=0
      NR=0
      DO 1 I=1,79
      IF(BLANK.NE.ALPHA(I))GO TO 2
    1 CONTINUE
    3 RETURN
    2 NL=I
      DO 4 I=1,79
      J=80-I
      IF(BLANK.NE.ALPHA(J)) GO TO 5
    4 CONTINUE
      GO TO 3
    5 NR=J
      GO TO 3
      END
```

```
      SUBROUTINE ULINE(BETA,NR,NL)
      DIMENSION BETA(80)
      DATA BANK/' '/
      DATA CRANK/'_'/
      NLM=NL-1
      NRP=NR+1
      IF(NLM.EQ.0)GO TO 2
      DO 1 I=1,NLM
      BETA(I)=BANK
    1 CONTINUE
    2 DO 3 I=NL,NR
      BETA(I)=CRANK
    3 CONTINUE
      IF(NRP.EQ.80) GO TO 4
      DO 5 I=NRP,79
      BETA(I)=BANK
    5 CONTINUE
    4 RETURN
      END
```

```
      SUBROUTINE CENTER(ALPHA,NR,NL)
      DIMENSION ALPHA(80),GAMMA(80)
      DATA TANK/' '/
      DO 8 I=1,79
      GAMMA(I)=ALPHA(I)
    8 CONTINUE
      NA=NR-NL+1
      NB=(79-NA)/2
      NC=NB-NL
      IF(NC)1,2,3
    2 RETURN
    3 J=79-NC
      DO 6 I=1,NC
      ALPHA(I)=TANK
    6 CONTINUE
      DO 5 I=1,J
      NE=NC+I
      ALPHA(NE)=GAMMA(I)
    5 CONTINUE
      GO TO 2
    1 NC=-NC
      J=79-NC
      DO 9 I=1,J
```

```
      NE=NC+I
      ALPHA(I)=GAMMA(NE)
    9 CONTINUE
      J=J+1
      DO 10 I=J,79
      ALPHA(I)=TANK
   10 CONTINUE
      GO TO 2
      END



      SUBROUTINE SPLINE(LINE)
      WRITE(6,1)
    1 FORMAT(1H )
      LINE=LINE+1
      RETURN
      END



      SUBROUTINE PAGENO(LROMAN,LINE)
      DIMENSION ROMAN(20)
      DATA PAGE/'PAGE'/
      DATA ROMAN/'I   ','II  ','III ','IV  ','V   ','VI  ','VII ',
     1'VIII','IX  ','X   ','XI  ','XII ','XIII','XIV ','XV  ','XVI ',
     2'XVII'/
      MAN=LROMAN-17
    1 FORMAT(1H0)
      IF(LROMAN.EQ.0) GO TO 5
      WRITE(6,1)
      IF(LROMAN.LT.18)WRITE(6,2)PAGE,ROMAN(LROMAN)
    2 FORMAT(50X,A4,1X,A4)
      IF(LROMAN.GE.18) WRITE(6,3) PAGE,MAN
    3 FORMAT(50X,A4,I4)
    5 LROMAN=LROMAN+1
      LINE=1
      WRITE(6,4)
    4 FORMAT('1'/////)
      RETURN
      END
```

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Picatinny Arsenal, Dover, New Jersey	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

EDITOR - A Computer Program for Documentation Purposes

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

William H. Bolte

6. REPORT DATE December 1968	7a. TOTAL NO. OF PAGES 28	7b. NO. OF REFS
8a. CONTRACT OR GRANT NO. b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) Technical Memorandum 1871 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	

10. DISTRIBUTION STATEMENT

Statement 1: Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT

This report describes the application of an editing program called "EDITOR" for documentation purposes.

This program was written in FORTRAN IV Language for the IBM 360/65 Digital Computer and requires a total memory length of 2[illegible],582 Bytes.

The "EDITOR" Program permits the user to have his reports "typed" on an IBM 1403 highspeed printer. All text editing instructions (line spacing, underlining, centering, page numbering, etc...) are passed to the "EDITOR" Program by means of Alpha Numerical codes punched in Column 1 of the input data cards. The data cards contain the text of the report, having been punched from FORTRAN coding sheets upon which the author has "written" his report.

DD FORM 1 NOV 65 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14 KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
EDITOR						
Documentation						
Computer Generated Text						

UNCLASSIFIED

Security Classification

AD683474

'EDITOR' MODIFICATION
Technical Memorandum #1871
Modified: 31 March 1969

CONTENTS: (1) "P" Code Description - Add to Section 4. "Code Description and Application"

(2) Program Listing Replacement - Replaces old TM 1871 Listing of Section 5. B. "Listing of FORTRAN Source Program."

WILLIAM H. BOLTE
SEAD/DPSO Bldg. 351
Picatinny Arsenal
Ext. 3663

CLEARINGHOUSE

PROGRAM: "EDITOR" - A computer program for documentation purposes (T.M. 1871)

MODIFIED: 31 March 1969

MODIFICATION: Added "P" Code

PURPOSE OF CODE: The "P" code allows one to skip a certain specified number of pages. This number is a user option.

INSTRUCTIONS FOR "P" CODE USAGE:

Punch the letter "P" in Column 1 of the card to be used, followed by an integer number. This number specifies the number of pages to be skipped and has an allowable field width of two (2); therefore, each data item must be right justified in its field, since leading and trailing blanks are treated as zeros (0's).

i.e.

	P	#	#
Col	1	2	3

NOTE: Any information punched after column three (3) is ignored and hence not printed during output.

ILLUSTRATIVE USE:

CODE - "P" - Multiple Page Skip

CARDS:

Printing on Card #3
P 2
L End of this page

RESULT:

END OF THIS PAGE
(Skips two (2) pages)

PRINTING ON CARD #3
(In this case it is printed on top of third (3rd) page)

COMMENT:

This modification sheet is to be used in accordance with the Technical Memorandum #1871. Any problems, questions, inquires or suggestions regarding the use of this code or any other subject matter from the T.M. 1871 should be directed to Mr. William H. Bolte, SEAD/DPSO, Building 351, Ext. 3563.

5.8. LISTING OF FORTRAN IV SOURCE PROGRAM

```
C                    'EDITOR' PROGRAM LISTING REVISION 31 MAR 69
C                         REPLACES TM#1871 PROGRAM LISTING


C       EDITOR PROGRAM, FOURTH VERSION, MARCH 1969        'P' CODE ADDED     A  50
C       TM = 1871 IS STILL TO BE USED AS REFERENCE                           A 100
        DIMENSION ALPHA(80), BETA(80), ROMAN(20)                             A 150
        DATA C/'C'/,U/'U'/,A/'A'/,XN/'N'/,R/'R'/,XL/'L'/,P/'P'/,PAGE/'PAGE   A 200
       1'/,X1/'1'/,X2/'2'/,X3/'3'/,X4/'4'/,X5/'5'/,X6/'6'/,X7/'7'/,X8/'8'/   A 250
       2,X9/'-'/,V/'V'/,SPACE/' '/,F/'F'/,S/'S'/                             A 300
1       LMAN=0                                                               A 350
        IA=0                                                                 A 400
        LINE=1                                                               A 450
        LROMAN=0                                                             A 500
        WRITE (6,22)                                                         A 550
2       READ(5,23,END=28) XC,(ALPHA(I),I=1,79)                               A 575
        IF (XC.EQ.V) IA=IA+1                                                 A 600
        IW=IA/2                                                              A 650
        IX=IA-2*IW                                                           A 700
        IF (IX.NE.1) GO TO 4                                                 A 750
        IF (XC.EQ.V) XC=SPACE                                                A 800
        BETA(1)=XC                                                           A 850
        DO 3 I=1,79                                                          A 900
        BETA(I+1)=ALPHA(I)                                                   A 950
3       CONTINUE                                                             A1000
        WRITE (6,26) (BETA(I),I=1,80)                                        A1050
        LINE=LINE+1                                                          A1100
        GO TO 13                                                             A1150
4       LSKIP=0                                                              A1200
        IF (XC.EQ.X1) LSKIP=1                                                A1250
        IF (XC.EQ.X2) LSKIP=2                                                A1300
        IF (XC.EQ.X3) LSKIP=3                                                A1350
        IF (XC.EQ.X4) LSKIP=4                                                A1400
        IF (XC.EQ.X5) LSKIP=5                                                A1450
        IF (XC.EQ.X6) LSKIP=6                                                A1500
        IF (XC.EQ.X7) LSKIP=7                                                A1550
        IF (XC.EQ.X8) LSKIP=8                                                A1600
        IF (XC.EQ.X9) LSKIP=9                                                A1650
        LSKIQ=LSKIP/2                                                        A1700
        LSKIR=LSKIP-2*LSKIQ                                                  A1750
        IF (LSKIR) 7,7,5                                                     A1800
5       DO 6 I=1,LSKIP                                                       A1850
```

```
      CALL SPLINE (LINE)                                                A1900
      IF (LINE.EQ.50) CALL PAGENO (LROMAN,LINE)                         A1950
6     CONTINUE                                                          A2000
7     IF (XC.EQ. ) CALL LIMIT (ALPHA,NR,NL)                             A2050
      IF (XC.EQ.U) CALL LIMIT (ALPHA,NR,NL)                             A2100
      IF (XC.EQ.A) CALL LIMIT (ALPHA,NR,NL)                             A2150
      IF (XC.EQ.C) CALL CENTER (ALPHA,NR,NL)                            A2200
      IF (XC.EQ.A) CALL CENTER (ALPHA,NR,NL)                            A2250
      IF (XC.EQ.A) CALL LIMIT (ALPHA,NR,NL)                             A2300
      IF (XC.EQ.U) CALL ULINE (BETA,NR,NL)                              A2350
      IF (XC.EQ.A) CALL ULINE (BETA,NR,NL)                              A2400
      IF (XC.EQ.U) GO TO 8                                              A2450
      IF (XC.EQ.A) GO TO 8                                              A2500
      IF (XC.EQ.XN) GO TO 8                                             A2550
      IF (XC.EQ.S) CALL SELECT (ALPHA,BETA)                             A2600
      IF (XC.EQ.S) GO TO 8                                              A2650
      IF (XC.EQ.F) GO TO 17                                             A2700
      WRITE (6,24) (ALPHA(I),I=1,79)                                    A2750
      LINE=LINE+2                                                       A2800
      GO TO 9                                                           A2850
8     WRITE (6,25) (ALPHA(I),I=1,79)                                    A2900
      LINE=LINE+1                                                       A2950
      IF (XC.EQ.U) WRITE (6,27) (BETA(I),I=1,79)                        A3000
      IF (XC.EQ.A) WRITE (6,27) (BETA(I),I=1,79)                        A3050
      IF (XC.EQ.S  WRITE (6,27) (BETA(I),I=1,79)                        A3100
      IF (XC.EQ.U) LINE=LINE+1                                          A3150
      IF (XC.EQ.A) LINE=LINE+1                                          A3200
      IF (XC.EQ.S) LINE=LINE+1                                          A3250
9     IF (LSKIP.EQ.0) GO TO 12                                          A3300
      IF (LSKIP) 12,10,12                                               A3350
10    DO 11 I=1,LSKIP                                                   A3400
      CALL SPLINE (LINE)                                                A3450
      IF (LINE.EQ.50) CALL PAGENO (LROMAN,LINE)                         A3500
11    CONTINUE                                                          A3550
12    IF (XC.EQ.R) GO TO 14                                             A3600
      IF (XC.EQ.F) GO TO 14                                             A3650
      IF (XC.EQ.XL) GO TO 14                                            A3700
13    IF (LINE.EQ.49) CALL SPLINE (LINE)                                A3750
      IF (LINE.EQ.50) CALL PAGENO (LROMAN,LINE)                         A3800
      GO TO 16                                                          A3850
 4    IF (LINE.EQ.50) GO TO 13                                          A3900
      INE=LINE+1                                                        A3950
      DO 15 I=INE,49                                                    A4000
      CALL SPLINE (LINE)                                                A4050
15    CONTINUE                                                          A4100
      GO TO 13                                                          A4150
16    IF (XC.EQ.F) GO TO 1                                              A4200
      IF (LMAN.EQ.1) GO TO 2                                            A4250
```

```
      IF (XC.EQ.R) LMAN=1                                               A4300
      IF (XC.EQ.R) LROMAN=18                                            A4350
      GO TO 2                                                           A4400
17    AA=ALPHA(1)                                                       A4450
      AB=ALPHA(2)                                                       A4500
      MA=0                                                              A4550
      IF (AA.EQ.X1) MA=1                                                A4600
      IF (AA.EQ.X2) MA=2                                                A4650
      IF (AA.EQ.X3) MA=3                                                A4700
      IF (AA.EQ.X4) MA=4                                                A4750
      IF (AA.EQ.X5) MA=5                                                A4800
      IF (AA.EQ.X6) MA=6                                                A4850
      IF (AA.EQ.X7) MA=7                                                A4900
      IF (AA.EQ.X8) MA=8                                                A4950
      IF (AA.EQ.X9) MA=9                                                A5000
      MB=0                                                              A5050
      IF (AB.EQ.X1) MB=1                                                A5100
      IF (AB.EQ.X2) MB=2                                                A5150
      IF (AB.EQ.X3) MB=3                                                A5200
      IF (AB.EQ.X4) MB=4                                                A5250
      IF (AB.EQ.X5) MB=5                                                A5300
      IF (AB.EQ.X6) MB=6                                                A5350
      IF (AB.EQ.X7) MB=7                                                A5400
      IF (AB.EQ.X8) MB=8                                                A5450
      IF (AB.EQ.X9) MB=9                                                A5500
      MC=10*MA+MB                                                       A5550
      IF (LINE.EQ.1) GO TO 19                                           A5600
18    IF (LINE.NE.50) CALL SPLINE (LINE)                                A5650
      IF (LINE.NE.50) GO TO 18                                          A5700
      IF (LINE.NE.50) CALL PAGENO (LROMAN,LINE)                         A5750
19    DO 21 I=1,MC                                                      A5800
      DO 20 J=1,49                                                      A5850
20    CALL SPLINE (LINE)                                                A5900
      CALL PAGENO (LROMAN,LINE)                                         A5950
21    CONTINUE                                                          A6000
      GO TO 2                                                           A6050
28    STOP                                                              A6100
C                                                                       A6150
22    FORMAT ('1'/////)                                                 A6200
23    FORMAT (80A1)                                                     A6250
24    FORMAT (17X,79A1,/)                                               A6300
25    FORMAT (17X,79A1)                                                 A6350
26    FORMAT (17X,80A1)                                                 A6400
27    FORMAT (1H+,16X,79A1,/)                                           A6450
      END                                                               A6500-
```

```
      SUBROUTINE SELECT (ALPHA,BETA)                                    B  50
      DIMENSION ALPHA(80), BETA(80)                                     B 100
      DATA RANK/' ',HANK/'-'/                                           B 150
      DO 1 I=1,79                                                       B 200
      IF (ALPHA(I).EQ.RANK) BETA(I)=RANK                                B 250
      IF (ALPHA(I).NE.RANK) BETA(I)=HANK                                B 300
1     CONTINUE                                                          B 350
      RETURN                                                            B 400
      END                                                               B 450-
```

```
      SUBROUTINE LIMIT (ALPHA,NR,NL)                                    C  50
      DIMENSION ALPHA(80)                                               C 100
      DATA BLANK/' '/                                                   C 150
      NL=0                                                              C 200
      NR=0                                                              C 250
      DO 1 I=1,79                                                       C 300
      IF (BLANK.NE.ALPHA(I)) GO TO 3                                    C 350
1     CONTINUE                                                          C 400
2     RETURN                                                            C 450
3     NL=I                                                              C 500
      DO 4 I=1,79                                                       C 550
      J=80-I                                                            C 600
      IF (BLANK.NE.ALPHA(J)) GO TO 5                                    C 650
4     CONTINUE                                                          C 700
      GO TO 2                                                           C 750
5     NR=J                                                              C 800
      GO TO 2                                                           C 850
      END                                                               C 900-
```

```
      SUBROUTINE ULINE (BETA,NR,NL)                                     D  50
      DIMENSION BETA(80)                                                D 100
      DATA BANK/' '/                                                    D 150
      DATA CRANK/'_'/                                                   D 200
      NLM=NL-1                                                          D 250
      NRP=NR+1                                                          D 300
      IF (NLM.EQ.0) GO TO 2                                             D 350
      DO 1 I=1,NLM                                                      D 400
      BETA(I)=BANK                                                      D 450
1     CONTINUE                                                          D 500
2     DO 3 I=NL,NR                                                      D 550
      BETA(I)=CRANK                                                     D 600
```

```
3     CONTINUE                                                          D 650
      IF (NRP.EQ.80) GO TO 5                                            D 700
      DO 4 I=NRP,79                                                     D 750
      BETA(I)=BANK                                                      D 800
4     CONTINUE                                                          D 850
5     RETURN                                                            D 900
      END                                                               D 950-


      SUBROUTINE CENTER (ALPHA,NR,NL)                                   E  50
      DIMENSION ALPHA(80), GAMMA(80)                                    E 100
      DATA TANK/' '/                                                    E 150
      DO 1 I=1,79                                                       E 200
      GAMMA(I)=ALPHA(I)                                                 E 250
1     CONTINUE                                                          E 300
      NA=NR-NL+1                                                        E 350
      NB=(79-NA)/2                                                      E 400
      NC=NB-NL                                                          E 450
      IF (NC) 6,2,3                                                     E 500
2     RETURN                                                            E 550
3     J=79-NC                                                           E 600
      DO 4 I=1,NC                                                       E 650
      ALPHA(I)=TANK                                                     E 700
4     CONTINUE                                                          E 750
      DO 5 I=1,J                                                        E 800
      NE=NC+I                                                           E 850
      ALPHA(NE)=GAMMA(I)                                                E 900
5     CONTINUE                                                          E 950
      GO TO 2                                                           E1000
6     NC=-NC                                                            E1050
      J=79-NC                                                           E1100
      DO 7 I=1,J                                                        E1150
      NE=NC+I                                                           E1200
      ALPHA(I)=GAMMA(NE)                                                E1250
7     CONTINUE                                                          E1300
      J=J+1                                                             E1350
      DO 8 I=J,79                                                       E1400
      ALPHA(I)=TANK                                                     E1450
8     CONTINUE                                                          E1500
      GO TO 2                                                           E1550
      END                                                               E1600-


      SUBROUTINE SPLINE (LINE)                                          F  50
```

```
      WRITE (6,1)                                                       F 100
      LINE=LINE+1                                                       F 150
      RETURN                                                            F 200
C                                                                       F 250
1     FORMAT (1H )                                                      F 300
      END                                                               F 350-


      SUBROUTINE PAGEND (LROMAN,LINE)                                   G  50
      DIMENSION ROMAN(20)                                               G 100
      DATA PAGE/'PAGE'/                                                 G 150
      DATA ROMAN/'I','II','III','IV','V','VI','VII','VIII','IX','X','XI' G 200
     1,'XII','XIII','XIV','XV','XVI','XVII'/                            G 250
      MAN=LROMAN-17                                                     G 300
      IF (LROMAN.EQ.0) GO TO 1                                          G 350
      WRITE (6,2)                                                       G 400
      IF (LROMAN.LT.18) WRITE (6,3) PAGE,ROMAN(LROMAN)                  G 450
      IF (LROMAN.GE.18) WRITE (6,4) PAGE,MAN                            G 500
1     LROMAN=LROMAN+1                                                   G 550
      LINE=1                                                            G 600
      WRITE (6,5)                                                       G 650
      RETURN                                                            G 700
C                                                                       G 750
2     FORMAT (1H0)                                                      G 800
3     FORMAT (50X,A4,1X,A4)                                             G 850
4     FORMAT (50X,A4,I4)                                                G 900
5     FORMAT ('1'//////)                                                G 950
      END                                                               G1000-
```