NOAA Eastern Region Computer Programs
and Problems NWS ERCP - No. 30

CERR: AN AVIATION VERIFICATION PROGRAM

Matthew R. Peroutka
National Weather Service Forecast Office
Cleveland, Ohio

Scientific Services Division
Eastern Region Headquarters
April 1985

U.S. DEPARTMENT OF COMMERCE / National Oceanic and Atmospheric Administration / National Weather Service

NOAA TECHNICAL MEMORANDUM

National Weather Service, Eastern Region Computer Programs and Problems

The Eastern Region Computer Programs and Problems (ERCP) series is a subset of the Eastern Region Technical Memorandum series. It will serve as the vehicle for the transfer of information about fully documented AFOS application programs. The format ERCP - No. 1 will serve as the model for future issuances in this series.

1 An AFOS version of the Flash Flood Checklist. Cynthia M. Scott, March 1981. (PB81 211252).

2 An AFOS Applications Program to Compute Three-Hourly Stream Stages. Alan P. Blackburn, September 1981. (PB82 156886).

3 PUPPY (AFOS Hydrologic Data Reporting Program). Daniel P. Provost, December 1981. (PB82 199720).

4 Special Search Computer Program. Alan P. Blackburn, April 1982. (PB83 175455).

5 Conversion of ALEMBIC$ Workbins. Alan P. Blackburn, October 1982. (PB83 138313).

6 Real-Time Quality Control of SAOs. John A. Billet, January 1983. (PB83 166082).

7 Automated Hourly Weather Collective from HRR Data Input. Lawrence Cedrone, January 1983 (PB83 167122).

8 Decoders for FRH, FTJ and FD Products. Cynthia M. Scott, February 1983. (PB83 176057).

9 Stability Analysis Program. Hugh M. Stone, March 1983. (PB83 197947).

10 Help for AFOS Message Comp. Alan P. Blackburn, May 1983. (PB83 213561).

11 Stability and Other Parameters from the First Transmission RAOB Data. Charles D. Little, May 1983. (PB83 220475).

12 TERR, PERR, and BIGC: Three Programs to Compute Verification Statistics. Matthew R. Peroutka, August 1983. (PB84 127521).

13 Decoder for Manually Digitized Radar Observations. Matthew R. Peroutka, June 1983. (PB84 127539).

14 Slick and Quick Data Entry for AFOS Era Verification (AEV) Program. Alan P. Blackburn, December 1983. (PB84 138726).

15 MDR--Processing Manually Digitized Radar Observations. Matthew R. Peroutka, November 1983. (PB84 161462)

16 RANP: Stability Analysis Program. Hugh M. Stone, February 1984.(PB84 161

17 ZONES. Gerald G. Rigdon, March 1984. (PB84 174325)

18 Automated Analysis of Upper Air Soundings to Specify Precipitation Type. Joseph R. Bocchieri and Gerald G. Rigdon, March 1984. (PB84 174333)

(Continued on Inside Rear Cover)

NOAA EASTERN REGION COMPUTER PROGRAMS AND PROBLEMS - No. 30

CERR: AN AVIATION VERIFICATION PROGRAM

Matthew R. Peroutka
National Weather Service Forecast Office
Cleveland, Ohio

Scientific Services Division
Eastern Region Headquarters
April 1985

CERR: AN AVIATION VERIFICATION PROGRAM

Matthew R. Peroutka
National Weather Service Forecast Office
Cleveland, Ohio

I. General Information

Aviation verification statistics have been computed for Cleveland forecasters since 1980. For most of these years, the computations were done by hand, and a simple statistic had to be developed. The CERR program was developed soon after the National Verification Programs began to run, and it has replaced all the manual computations. It may be run in addition to the Western Region AEV program (Dunn, 1985) to provide a different view of the results.

The statistic chosen for verification was the average MOS category error. This statistic is simply the absolute value of the difference between the category forecast and the category observed. Scores are computed for each forecaster and guidance at each station for each period forecast. The percent improvement over guidance is also computed.

The scores are computed for ceiling and visibility forecasts.

II. Methodology and Software Structure

The program is written in Data General's FORTRAN IV, and runs in the background partition of a WSFO system. The verification data files VERDIR and VDATA should reside in the current directory or be accessible through links.

The program begins with a dialogue with the operator at the system console. The Appendix contains a sample dialogue. Output from the program is appended to a file named VOUT. If the file does not exist, it is created. This is the same technique used by three other verification programs at the Cleveland WSFO (see Peroutka, 1983). Sending the statistics to a file provides a great deal of flexibility. The data can be archived in any manner desired; hard copy can be produced at a convenient time from the Dasher or PPM. Smaller files can be displayed at an AFOS ADM using the DSP: command.

Since the output from each run is appended to the previous file, the program can be executed a number of times before a report is printed.

A sample report appears in Figure 3.

The program file requires about 40 disk blocks. About 2ØK of memory is required to execute. A report for ten forecasters for an entire year takes less than a minute if the program file resides on a cartridge drive.

III. Cautions and Restrictions

If CERR is run on a sparse VDATA file (e.g. some forecast and/or observations categories are empty (Ø)), runtime errors may result.

IV. References

1. Dunn, Lawrence B.. 1985: AEV Local Verification for Aviation, Precipitation and Temperature Programs. National Weather Service Western Region Computer Programs and Problems Number 42.

2. Heffernan, Mary and Mary Newton. 1982: AFOS-ERA Forecast Verification Program. National Weather Service Techniques Development Laboratory CP 82-2.

3. Peroutka, Matthew R. 1983: TERR, PERR, and BIGC: Three Programs to Computer Verification Statistics. National Weather Service Eastern Region Computer Programs and Problems Number 12 (PB 84-127521).

V. Program Information and Procedures

CERR: AN AVIATION VERIFICATION PROGRAM

Part A: INFORMATION AND INSTALLATION

PROGRAM NAME: CERR.SV

AAL ID:
REV NO.: 1.00

PURPOSE: To compute aviation verification statistics.

PROGRAM INFORMATION:

Development Programmer:
Matthew R. Peroutka
Location: WSFO Cleveland, OH
Phone: FTS 942-4949
Language: FORTRAN IV/Rev. 5.57
Save File Creation Dates: 4/29/85
Running Time: One to five minutes
Disk Space: Program Files - 40 RDOS Blocks
Data Files - 366 RDOS Blocks

Maintenance Programmer:
Matthew R. Peroutka
Location: WSFO Cleveland, OH
Phone: FTS 942-4949
Type: Normal

PROGRAM REQUIREMENTS:

Program Files:

Name	Comments
CERR.SV	Main program

Data Files:

Name	DP Location	Read/Write	Comments
VDATA	DPØ or DPØF	Read	Raw verification data
VERDIR	DPØ or DPØF	Read	Station information
VOUT	DPØ	Write	Output File.

LOAD LINE

```
RLDR CERR CEIN CEØ CEUP IC CEEC CER1 JDATE CERPT J2MDA CECOM
     CERC <BG UTIL FORT>.LB
```

PROGRAM INSTALLATION

1. Move CERR.SV onto the disk and set up any necessary links.

CERR: AN AVIATION VERIFICATION PROGRAM

PART B: EXECUTION AND ERROR CONDITIONS

PROGRAM NAME: CERR.SV

AAL ID:
REV No.: 1.00

PROGRAM EXECUTION:

1. Answer questions as they are asked at the system console (see Appendix of ERCP #30).

2. Print or display the file VOUT when program is complete.

ERROR CONDITIONS

ADM MESSAGES	MEANING
None	

DASHER MESSAGES	MEANING
None	

V. Figures

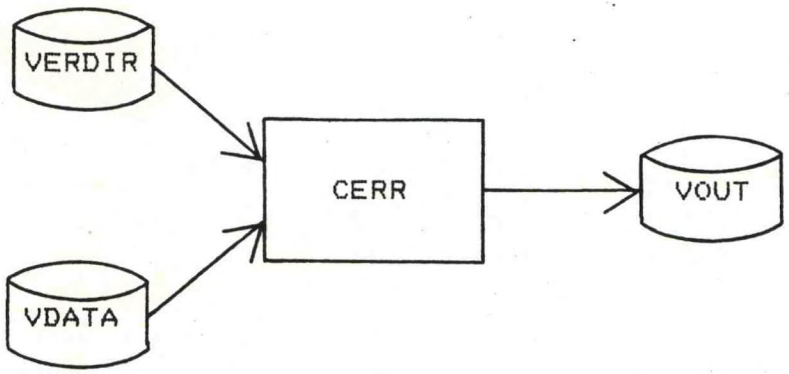

Figure 1. Data flow for CERR.SV.

MAIN PROGRAM
CERR

SUBROUTINES
CEIN
CE0
CEUP → IC, CEEC
JDATE
CER1
CERPT → J2MDA, CECOM, CERC

Figure 2. Software Structure for CERR.SV.

MEAN AVIATION CATEGORY ERROR

FORECASTER 8: 5/ 4 TO 7/30 1984 15 FORECASTS

CEILINGS		CLE			CMH			CVG		
1ST PD	YOU:	0.2			0.2			0.1		
YOU/MOS (% IMPR):		0.2/	0.4	(50.)	0.2/	0.2	(0.)	0.1/	0.3	(60.)
2ND PD	YOU:	0.1			0.5			0.5		
YOU/MOS (% IMPR):		0.1/	0.1	(0.)	0.5/	0.6	(11.)	0.5/	0.4	(-33.)
3RD PD	YOU:	0.1			0.5			0.5		
YOU/MOS (% IMPR):		0.1/	0.1	(0.)	0.5/	0.4	(-17.)	0.5/	0.5	(0.)
CITY AVG (% IMPR):		0.2		(17.)	0.4		(-2.)	0.4		(9.)
AVG (% IMPR):		0.3		(8.)						

VISIBILITIES		CLE			CMH			CVG		
1ST PD	YOU:	0.4			0.9			0.4		
YOU/MOS (% IMPR):		0.4/	0.7	(45.)	0.9/	1.3	(30.)	0.4/	0.6	(36.)
2ND PD	YOU:	0.0			0.4			0.6		
YOU/MOS (% IMPR):		0.0/	0.1	(100.)	0.4/	0.6	(33.)	0.6/	0.6	(0.)
3RD PD	YOU:	0.4			0.4			0.4		
YOU/MOS (% IMPR):		0.4/	0.4	(17.)	0.4/	0.5	(14.)	0.4/	0.5	(22.)
CITY AVG (% IMPR):		0.3		(54.)	0.6		(26.)	0.5		(20.)
AVG (% IMPR):		0.4		(33.)						

MEAN AVIATION CATEGORY ERROR

FORECASTER 9: 7/ 3 TO 7/14 1984 4 FORECASTS

CEILINGS		CLE			CMH			CVG		
1ST PD	YOU:	0.5			0.2			0.0		
YOU/MOS (% IMPR):		0.5/	0.5	(0.)	0.2/	0.2	(0.)	0.0/	0.0	(0.)
2ND PD	YOU:	0.2			0.5			0.7		
YOU/MOS (% IMPR):		0.2/	0.2	(0.)	0.5/	0.5	(0.)	0.7/	0.5	(-50.)
3RD PD	YOU:	0.0			0.5			1.5		
YOU/MOS (% IMPR):		0.0/	0.2	(100.)	0.5/	0.7	(33.)	1.5/	1.0	(-50.)
CITY AVG (% IMPR):		0.2		(33.)	0.4		(11.)	0.7		(-33.)
AVG (% IMPR):		0.5		(4.)						

VISIBILITIES		CLE			CMH			CVG		
1ST PD	YOU:	0.5			0.0			0.3		
YOU/MOS (% IMPR):		0.5/	0.5	(0.)	0.0/	0.5	(100.)	0.3/	0.0	(0.)
2ND PD	YOU:	0.5			0.5			0.5		
YOU/MOS (% IMPR):		0.5/	0.5	(0.)	0.5/	0.7	(33.)	0.5/	0.5	(0.)
3RD PD	YOU:	0.2			0.7			0.2		
YOU/MOS (% IMPR):		0.2/	0.5	(50.)	0.7/	1.0	(25.)	0.2/	0.7	(67.)
CITY AVG (% IMPR):		0.4		(17.)	0.4		(53.)	0.4		(22.)
AVG (% IMPR):		0.4		(31.)						

Figure 3. Sample VOUT output

VI. Appendix

```
OP3:CERR
 MONTH TO BEGIN (1-12)?  1
  MONTH TO STOP (1-12)?  8
 FROM WHAT YEAR DID THIS DATA COME?  1984
 HOW MANY FORECASTERS (1-30)?  4
 THEIR ID NUMBERS?  3,8,9,19
 OUTPUT WILL BE APPENDED TO A FILE NAMED VOUT.
 YOU CAN PRINT OR TYPE IT WHEN YOU ARE DONE.
R
```

A sample dialogue is displayed above. The program's responses are underlined for clarity. The first two questions allow the operator to determine which months of the year to process. The minimum period is one month and the maximum is one year. The year is used to determine file locations during leap years, and it is printed in the output file. There is no checking between the data in the VDATA file and the year input.

The next two questions deal with which forecasters to process. The program can handle from one to thirty forecasters in any given run. (It is much more efficient to let the program handle the forecasters as a group than to make many individual runs.) Be sure the number of forecasters and the number of id's entered agree.

Since the output file VOUT can become quite long, it is wise to delete it once the data is no longer needed. See Figure 3 for sample output.

CERR.SV will issue a report for each forecaster whose id is input. If more than one forecaster id is given, then a collective report is generated at the end of the file.

The first line of the report gives the forecaster's id number, the dates of the first and last forecasts processed, and the number of forecasts processed. The next line contains headings for the stations verified. Ceilings and visibilities are verified in two separate tables.

The first line of each table contains the mean category error for this forecaster for each station in the first period. The next line has three values for each station. To the left of the slash is the forecaster's mean category error (the same as above), but it only includes cases when a MOS forecast was available. To the right of the slash is the mean category error for MOS. In parentheses, following these two numbers is the percent improvement that the forecaster made over MOS.

The next four lines contain similar information for the second and third period forecasts.

Line nine contains the average over three periods for each station. The forecaster's mean category error and improvement over MOS are both computed. The final line of the report contains the same two statistics averaged over all stations.

The collective report uses the same format, but the word ALL replaces the forecaster number.

VII. Program Listings

```
C            CERR.SV                        AVIATION CATGORY ERRORS
      DIMENSION ISTNS(2,3), IFRI(30), IBF(256), I12LA(30,2), ICE(12)
      DIMENSION ICERR(30,3,3,3), IVERR(30,3,3,3)
      DIMENSION NUMC(30,3,3,3), NUMV(30,3,3,3)
C
C      THIS PROGRAM WILL GENERATE A REPORT ON MEAN ABSOLUTE AVIATION CATAGORY
C      ERRORS FOR A GROUP OF FORECASTERS AND MOS.  INPUT COMES FROM THE DASHER,
C      AND THE FILES VDATA AND VERDIR.  OUTPUT IS APPENDED TO A FILE NAMED VOUT.C
C      RLDR CERR CEIN CE0 CEUP IC CEEC CER1 JDATE CERPT J2MDA CECOM CERC↑
C       <BG UTIL FORT AFOSE>.LB
C
C      ARRAY DIMENSIONS FOR ICERR, IVERR, NUMC, AND NUMV:
C
C      ICERR(I,J,K,L)      I = FCSTR NUMBER.  IFRI(I) IS THE ID FOR THIS FCSTR.
C                          J = PERIOD.  1 = 12HR, 2 = 18HR, 3 = 24HR.
C                          K = STATION.  ISTNS(1,K) AND ISTNS(2,K) CONTAIN ID.
C                          L = TYPE.  1 = FCSTR, 2 = FCSTR VS. MOS, 3 = MOS.
C
      CALL CEIN(MB,MS,IY,NFR,IFRI,NSTA,ISTNS)       ;GET INPUT DATA.
      CALL CE0(NFR,NSTA,ICERR,IVERR,NUMC,NUMV,I12LA);ZERO ARRAYS.
      CALL GCHN(ICH,IER)                            ;OPEN FILE.
      CALL OPENR(ICH,"VDATA",0,IER)
      IR1 = JDATE(MB,1,IY) - 1
      MS = MS + 1
      IR2 = JDATE(MS,1,IY) - 2
      IF (MS.EQ.13) IR2 = JDATE(12,31,IY) - 1
      DO 500 I = IR1,IR2                            ;LOOP TO CULL DATA FROM
       CALL RDB(ICH,IBF,I,1,IER)                    ;THE FILE.
       DO 500 J = 1,NSTA
        DO 500 K1 = 1,13,12
         K = K1 - 1
         CALL CEUP(ISTNS(1,J),ISTNS(2,J),K,IBF,NFR,IFRI,IFN,ICE)
         IF (IFN.EQ.-1) GOTO 500
         DO 200 L = 1,3                                 ;LOOP FOR CIG ERRS.
          CALL CER1(IFN,L,J,ICERR,NUMC,ICE)
  200    CONTINUE
         DO 400 L = 7,9                                 ;LOOP FOR VSBY ERRS.
          CALL CER1(IFN,L,J,IVERR,NUMV,ICE)
          IF (I12LA(IFN,1).EQ.0) I12LA(IFN,1) = I+1
          I12LA(IFN,2) = I+1
  400    CONTINUE
  500    CONTINUE
      CALL KLOSE(ICH,IER)
      ICH = 1
      CALL APPEND(ICH,"VOUT",0,IER)
      DO 600 I = 1,NFR
       CALL CERPT(ICH,I,IFRI,NFR,I12LA,IY,NUMC,NUMV,NSTA,ISTNS,
     + ICERR,IVERR)
  600 CONTINUE
      CALL CERPT(ICH,31,IFRI,NFR,I12LA,IY,NUMC,NUMV,NSTA,ISTNS,
     +ICERR,IVERR)
      CALL CLOSE(ICH,IER)
      STOP
      END
```

```
      SUBROUTINE CEIN(MB,MS,IY,NFR,IFRI,NSTA,ISTNS)
      DIMENSION IBF(10),IFRI(30),ISTNS(2,3)
C
C      THIS SUBROUTINE RUNS THE INITIAL DIALOG FOR CERR.SV IT ALSO PULLS DATA
C      FROM THE VERDIR FILE.
C
  100 ACCEPT " MONTH TO BEGIN (1-12)?  ",MB
      IF (MB.GT.12.OR.MB.LT.1) GOTO 200
      ACCEPT " MONTH TO STOP (1-12)?  ",MS
      IF (MS.GT.12.OR.MS.LT.1) GOTO 200
      IF (MS.LT.MB) GOTO 200
      GOTO 300
  200 TYPE " INVALID MONTH"
      GOTO 100
  300 ACCEPT " FROM WHAT YEAR DID THIS DATA COME?  ",IY
      ACCEPT " HOW MANY FORECASTERS (1-30)?  ",NFR
      ACCEPT " THEIR ID NUMBERS?  ",(IFRI(I),I=1,NFR)
      CALL GCHN(IC,IER)
      CALL OPENR(IC,"VERDIR",0,IER)
      CALL RDS(IC,IBF,20,IER)
      CALL KLOSE(IC,IER)
      NSTA = IBF(3)
      IF (NSTA.GT.3) NSTA = 3
      DO 400 I = 1,NSTA
       J = 3+2*I
       ISTNS(1,I) = IBF(J)
       ISTNS(2,I) = IBF(J+1)
  400 CONTINUE
      TYPE " OUTPUT WILL BE APPENDED TO A FILE NAMED VOUT."
      TYPE " YOU CAN PRINT OR TYPE IT WHEN YOU ARE DONE."
      RETURN
      END
```

```
      SUBROUTINE CE0(NFR,NSTA,ICERR,IVERR,NUMC,NUMV,I12LA)
      DIMENSION ICERR(30,3,3,3),IVERR(30,3,3,3),I12LA(30,2)
      DIMENSION NUMC(30,3,3,3), NUMV(30,3,3,3)
C
C      THIS SUBROUTINE WILL ZERO THE ARRAYS.
C
      DO 500 I = 1,NFR
       DO 500 J = 1,3
        DO 500 K = 1,NSTA
         DO 500 L = 1,3
          ICERR(I,J,K,L) = 0
          IVERR(I,J,K,L) = 0
          NUMC(I,J,K,L) = 0
          NUMV(I,J,K,L) = 0
  500 CONTINUE
      DO 700 I = 1,NFR
  700 I12LA(I,1) = 0
      RETURN
      END
```

```
      SUBROUTINE CEUP(ISTN1,ISTN2,ICY,IBF,NFR,IFRI,IFN,ICE)
      DIMENSION IBF(256), IFRI(30), ICE(12), ICATS(18)
C
C      THIS SUBROUTINE WILL SEARCH IBF FOR THE BLOCK FOR THE STATION IN
C      (ISTN1,ISTN2) AND THE CYCLE ICY.  THE TWELVE ABSOLUTE CIG ERRORS ARE
C      LOADED INTO ICE, MOS THEN LOCAL, FOLLOWED BY VSBYS.  IF AN ERROR IS
C      UNAVAILABLE, IT WILL EQUAL -1.  THE FORECASTER ID IS COMPARED WITH
C      THE GROUP AND A POSITION IS RETURNED IN IFN.  IF THE FORECAST OR
C      THE FORECASTER NUMBER ARE NOT FOUND, IFN WILL EQUAL -1.
C
      DO 100 JOF = 1,241,40                              ;FIND THE BLOCK OFFSET.
       IOF = JOF - 1
       IF (IBF(IOF+1).NE.ISTN1) GOTO 100
       IF (ISHFT(IBF(IOF+2),-8).NE.ISHFT(ISTN2,-8)) GOTO 100
       J = IBF(IOF+3)
       J = J - J/100*100
       IF (J.EQ.ICY) GOTO 200
  100 CONTINUE
      IFN = -1                                           ;UNSUCCESSFUL SEARCH.
      RETURN
  200 J = IBF(IOF+4) - (IBF(IOF+4)/100*100)              ;AVIATION FORECASTER NUMBER.
      DO 300 I = 1,NFR
      IFN = I
       IF (J.EQ.IFRI(I)) GOTO 400
  300 CONTINUE
      IFN = -1                                           ;UNSUCCESSFUL SEARCH.
      RETURN
  400 DO 420 I = 1,2                                     ;UNPACK FORECASTS INTO
       J = 9*(I-1) + 1                                   ;ICATS ARRAY.
       K = IBF(IOF+(I-1)*5+19)                           ;MOS CIGS AND VSBYS.
       ICATS(J) = ISHFT(IAND(K,170000K),-12)
       ICATS(J+1) = ISHFT(IAND(K,360K),-4)
       ICATS(J+2) = IAND(K,17K)
  420 CONTINUE
      DO 440 I = 1,2                                     ;UNPACK LOCAL AND OBSERVED
       J = (I-1)*3 + 4                                   ;CEILINGS.
       K = IOF + (I-1)*2 + 20
       ICATS(J) = IC(ISHFT(IBF(K),-8),1)
       L = IBF(K+1)
       ICATS(J+1) = IC(ISHFT(L,-8),1)
       ICATS(J+2) = IC(IAND(L,377K),1)
  440 CONTINUE
      DO 460 I = 1,2                                     ;UNPACK LOCAL AND OBSERVED
       J = (I-1)*3 + 13                                  ;VISIBILITIES.
       K = IOF + (I-1)*2 + 25
       ICATS(J) = IC(ISHFT(IBF(K),-8),2)
       L = IBF(K+1)
       ICATS(J+1) = IC(ISHFT(L,-8),2)
       ICATS(J+2) = IC(IAND(L,377K),2)
  460 CONTINUE
      CALL CEEC(ICATS,ICE)                               ;COMPUTE ERRORS.
      RETURN
      END
```

```
      SUBROUTINE CER1(IFN,L,J,ICARR,NUMAR,ICE)
      DIMENSION ICARR(30,3,3,3), NUMAR(30,3,3,3), ICE(12)
C
C      THIS SUBROUTINE ACCUMULATES THE ERRORS FROM THE ICE ARRAY INTO THE
C      ICERR, NUMC, IVERR, AND NUMV ARRAYS.
C
      L1 = L
      IF (L.GT.3) L1 = L - 6
      IF (ICE(L+3).EQ.-1) GOTO 100                        ;FORECASTER ERROR.
      ICARR(IFN,L1,J,1) = ICARR(IFN,L1,J,1) + ICE(L+3)
      NUMAR(IFN,L1,J,1) = NUMAR(IFN,L1,J,1) + 1
      IF (ICE(L).EQ.-1) RETURN                            ;FORECASTER V MOS
      ICARR(IFN,L1,J,2) = ICARR(IFN,L1,J,2) + ICE(L+3) ;ERROR.
      NUMAR(IFN,L1,J,2) = NUMAR(IFN,L1,J,2) + 1
   50 ICARR(IFN,L1,J,3) = ICARR(IFN,L1,J,3) + ICE(L)    ;MOS ERROR.
      NUMAR(IFN,L1,J,3) = NUMAR(IFN,L1,J,3) + 1
      RETURN
  100 IF (ICE(L).EQ.-1) RETURN
      GOTO 50
      END
```

```
      INTEGER FUNCTION IC(IVAL,ITYPE)
C
C       THIS FUNCTION WILL CONVERT CEILINGS AND VISIBILITIES INTO MOS CATAGOR-
C       IES.  THE VALUE OF THE CEILING OR VISIBILITY IS STORED IN IVAL.  ITYPE
C       IS ONE FOR A CEILING AND TWO FOR A VISIBILITY.  IF IVAL CONTAINS 377
C       OCTAL (ALL ONE BITS IN THE RIGHT BYTE), THEN A CATAGORY OF -1 IS RE-
C       TURNED.
C
      IF (IVAL.NE.377K) GOTO 100
       IC = -1
       RETURN
  100 IF (ITYPE.EQ.2) GOTO 500
      IF (IVAL.GE.2) GOTO 150                          ;CEILING SECTION.
       IC = 1
       RETURN
  150 IF (IVAL.GE.5) GOTO 200
       IC = 2
       RETURN
  200 IF (IVAL.GE.10) GOTO 250
       IC = 3
       RETURN
  250 IF (IVAL.GE.30) GOTO 300
       IC = 4
       RETURN
  300 IF (IVAL.GT.75) GOTO 350
       IC = 5
       RETURN
  350 IC = 6
      RETURN
  500 IF (IVAL.GE.2) GOTO 550                          ;VISIBILITY SECTION.
       IC = 1
       RETURN
  550 IF (IVAL.GE.10) GOTO 600
       IC = 2
       RETURN
  600 IF (IVAL.GE.30) GOTO 650
       IC = 3
       RETURN
  650 IF (IVAL.GE.50) GOTO 700
       IC = 4
       RETURN
  700 IF (IVAL.GE.70) GOTO 750
       IC = 5
       RETURN
  750 IC = 6
      RETURN
      END
```

```
      SUBROUTINE CEEC(ICATS,ICE)
      DIMENSION ICATS(18), ICE(12)
C
C       THIS SUBROUTINE USES THE CATGORIES IN ICATS TO COMPUTE CATAGORY
C       ERRORS WHICH ARE THEN LOADED INTO ICE.
C
      DO 300 I = 1,3
       I3 = I + 3
       I6 = I + 6
       J = I + 9
       J3 = J + 3
       J6 = J + 6
       IF (ICATS(I6).NE.377K) GOTO 100              ;CEILING ERRORS.
        ICE(I) = -1
        ICE(I3) = -1
        GOTO 200
  100  IF (ICATS(I).NE.17K) GOTO 120
        ICE(I) = -1
        GOTO 140
  120  ICE(I) = IABS(ICATS(I) - ICATS(I6))
  140  IF (ICATS(I3).NE.377K) GOTO 160
        ICE(I3) = -1
        GOTO 200
  160  ICE(I3) = IABS(ICATS(I3) - ICATS(I6))
  200  IF (ICATS(J6).NE.377K) GOTO 220              ;VISIBILITY ERRORS.
        ICE(I6) = -1
        ICE(J) = -1
        GOTO 300
  220  IF (ICATS(J).NE.17K) GOTO 240
        ICE(I6) = -1
        GOTO 260
  240  ICE(I6) = IABS(ICATS(J) - ICATS(J6))
  260  IF (ICATS(J3).NE.377K) GOTO 280
        ICE(J) = -1
        GOTO 300
  280  ICE(J) = IABS(ICATS(J3) - ICATS(J6))
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CERPT(IC,I,IFRI,NFR,I12LA,IY,NUMC,NUMV,NSTA,ISTNS,
     +ICERR,IVERR)
      DIMENSION I12LA(30,2),NUMC(30,3,3,3),NUMV(30,3,3,3),ISTNS(2,3)
      DIMENSION IFRI(30),ICERR(30,3,3,3),IVERR(30,3,3,3),ERC(3,3,4)
      DIMENSION ERV(3,3,4),IPD(2,3)
C
C      THIS SUBROUTINE WILL WRITE A REPORT TO CHANNEL IC FOR FORECASTER
C      I IN THE ARRAY IFRI.  IF I EQUALS 31, A COLLECTIVE REPORT IS WRITTEN.
C
      IF (I.NE.31) GOTO 200
      IF (NFR.EQ.1) RETURN                    ;NO COLLECTIVES FOR ONE FORECASTER.
      CALL CECOM(IC,NFR,I12LA,IY,NUMC,NUMV,NSTA,ICERR,IVERR)
      I = 1
      GOTO 400
  200 WRITE (IC,900)                          ;SECTION FOR INDIVIDUAL REPORTS.
      CALL J2MDA(I12LA(I,1),M1,ID1,IY)        ;GET FIRST AND LAST DATES.
      CALL J2MDA(I12LA(I,2),M2,ID2,IY)
      WRITE (IC,920) IFRI(I),M1,ID1,M2,ID2,IY,NUMC(I,1,1,1)
  400 WRITE (IC,930) (ISTNS(1,J),ISTNS(2,J),J=1,NSTA)    ;STATION ID'S.
      IPD(1,1) = "1S"                         ;LOAD PERIOD ARRAY.
      IPD(2,1) = "T "
      IPD(1,2) = "2N"
      IPD(2,2) = "D "
      IPD(1,3) = "3R"
      IPD(2,3) = "D "
      DO 600 J = 1, 3                         ;FILL ERC AND ERV WITH SCORES.
      DO 600 K = 1, NSTA
      DO 500 L = 1, 3
       ERC(J,K,L) = FLOAT(ICERR(I,J,K,L))/FLOAT(NUMC(I,J,K,L))
       ERV(J,K,L) = FLOAT(IVERR(I,J,K,L))/FLOAT(NUMV(I,J,K,L))
  500  CONTINUE
       ERC(J,K,4) = (ERC(J,K,3) - ERC(J,K,2))/ERC(J,K,3) * 100.
       ERV(J,K,4) = (ERV(J,K,3) - ERV(J,K,2))/ERV(J,K,3) * 100.
  600 CONTINUE
      CALL CERC(IC,NSTA,IPD,ERC)              ;WRITE CEILING SECTION.
      WRITE (IC,940) (ISTNS(1,J),ISTNS(2,J),J=1,NSTA)    ;STATION IDS.
      CALL CERC(IC,NSTA,IPD,ERV)              ;WRITE VSBY SECTION.
      RETURN
  900 FORMAT(/"<12>",1X/"<12>",1X/"<12>",1X/"<12>",1X/"<12>",18X,
     +"MEAN AVIATION CATEGORY ERROR"/"<12>",1X/"<12>",1X/)
  920 FORMAT("<12>"," FORECASTER",I4,":",4X,I4,"/",I2," TO",I3,"/",
     +I2,I5,4X,I3," FORECASTS"/"<12>",1X)
  930 FORMAT("<12>CEILINGS",14X,3(A2,A1,16X))
  940 FORMAT("<12>"/"<12>VISIBILITIES",10X,3(A2,A1,16X))
      END
```

```
      SUBROUTINE CECOM(IC,NFR,I12LA,IY,NUMC,NUMV,NSTA,ICERR,IVERR)
      DIMENSION I12LA(30,2),NUMC(30,3,3,3),NUMV(30,3,3,3)
      DIMENSION ICERR(30,3,3,3),IVERR(30,3,3,3)
C
C       THIS SUBROUTINE COMBINES THE ERROR DATA FOR ALL FORCASTERS SO
C       CERPT CAN WRITE A COLLECTIVE REPORT.
C
      I1 = I12LA(1,1)
      ILA = I12LA(1,2)
      DO 100 J = 2,NFR
       IF (I12LA(J,1).GT.0.AND.I12LA(J,1).LT.I1) I1 = I12LA(J,1)
       IF (I12LA(J,2).GT.0.AND.I12LA(J,1).GT.ILA) ILA = I12LA(J,2)
       DO 100 K = 1,3                          ;K COUNTS PERIODS.
        DO 100 L = 1,NSTA                      ;L COUNTS STATIONS.
         DO 100 M = 1,3                        ;M COUNTS TYPES.
          ICERR(1,K,L,M) = ICERR(1,K,L,M) + ICERR(J,K,L,M)
          IVERR(1,K,L,M) = IVERR(1,K,L,M) + IVERR(J,K,L,M)
          NUMC(1,K,L,M) = NUMC(1,K,L,M) + NUMC(J,K,L,M)
          NUMV(1,K,L,M) = NUMV(1,K,L,M) + NUMV(J,K,L,M)
  100 CONTINUE
      WRITE (IC,900)                           ;HEADING.
      CALL J2MDA(I1,M1,ID1,IY)                 ;GET FIRST AND LAST DATES.
      CALL J2MDA(ILA,M2,ID2,IY)
      WRITE (IC,910) M1,ID1,M2,ID2,IY,NUMC(1,1,1,1)
      RETURN
  900 FORMAT(/"<12>",1X/"<12>",1X/"<12>",1X/"<12>",1X/"<12>",18X,
     +"MEAN AVIATION CATEGORY ERROR"/"<12>",1X/"<12>",1X/)
  910 FORMAT("<12>"," FORECASTER ALL:",4X,I4,"/",I2," TO",I3,"/",
     +I2,I5,4X,I3," FORECASTS"/"<12>",1X)
      END
```

```
      SUBROUTINE CERC(IC,NSTA,IPD,ERC)
      DIMENSION ERC(3,3,4),IPD(2,3)
C
C       THIS SUBROUTINE WRITE THE CIG & VSBY REPORT TO CHANNEL IC FOR CERPT
C
      DO 700 J = 1, 3                  ;WRITE SCORES TO FILE.
       WRITE (IC,940) IPD(1,J), IPD(2,J), (ERC(J,K,1),K=1,NSTA)
       WRITE (IC,950) ((ERC(J,K,L),L=2,4),K=1,NSTA)
  700 CONTINUE
      TERR = 0.                        ;CITY AVERAGES AND TOTAL SCORES.
      TIMP = 0.
      DO 800 J = 1, NSTA
       ERC(1,J,1) = (ERC(1,J,1) + ERC(2,J,1) + ERC(3,J,1))/3.
       ERC(1,J,4) = (ERC(1,J,4) + ERC(2,J,4) + ERC(3,J,4))/3.
       TERR = TERR + ERC(1,J,1)
       TIMP = TIMP + ERC(1,J,4)
  800 CONTINUE
      TERR = TERR/FLOAT(NSTA)
      TIMP = TIMP/FLOAT(NSTA)
      WRITE(IC,960) (ERC(1,J,1),ERC(1,J,4),J=1,NSTA)
      WRITE(IC,970) TERR,TIMP
      RETURN
  940 FORMAT("<12>",1X,2A2,"PD",8X,"YOU:  ",3(F4.1,15X))
  950 FORMAT("<12>","  YOU/MOS (% IMPR):  ",3(F4.1,"/",F4.1," (",F5.0,
     +")  "))
  960 FORMAT("<12> CITY AVG (% IMPR):  ",3(F4.1,6X,"(",F5.0,")  "))
  970 FORMAT("<12>",6X,"AVG (% IMPR):  ",F4.1,6X,"(",F5.0,")")
      END
```

```
      INTEGER FUNCTION JDATE(MONTH,IDAY,IYEAR)
C
C      THIS FUNCTION WILL GENERATE A JULIAN DATE FOR THE MONTH, DAY, AND
C      YEAR INPUT.  IF THERE IS A PROBLEM WITH THE INPUT DATA, THE FUNC-
C      TION WILL RETURN A VALUE OF ZERO.
C
      COMMON /QJULQ/ MLIST(12)
      DATA MLIST/31,28,31,30,31,30,31,31,30,31,30,31/
      JDATE = 0                              ;CHECK INPUT DATA.
      IF (MONTH.LT.1.OR.MONTH.GT.12) RETURN
      IF (IYEAR.LT.1) RETURN
      IF (IDAY.LT.1) RETURN
      MLIST(2) = 28
      I = IYEAR/4*4                          ;IS THIS A LEAP YEAR?
      IF (I.EQ.IYEAR) MLIST(2) = 29
      I = IYEAR/100*100
      IF (I.EQ.IYEAR) MLIST(2) = 28
      I = IYEAR/400*400
      IF (I.EQ.IYEAR) MLIST(2) = 29
      IF (IDAY.GT.MLIST(MONTH)) RETURN
      DO 100 I = 1,MONTH                     ;COMPUTE JULIAN DATE.
  100 JDATE = JDATE + MLIST(I)
      JDATE = JDATE - MLIST(MONTH) + IDAY
      RETURN
      END
```

```
      SUBROUTINE J2MDA(JDA,M,ID,IY)
      DIMENSION MS(12)
C
C       THIS SUBROUTINE ACCEPTS A JULIAN DATE (JDA) AND A YEAR (IY).  IT
C       PRODUCES A MONTH (M) AND A DAY (ID).  IF THE JULIAN DATE IS BAD,
C       M AND ID WILL BOTH EQUAL ZERO.
C
      M = 0
      ID = 0
      IF (JDA.LE.0) RETURN                  ;CHECK FOR BAD JDA.
      MS(1) = 31                            ;SET UP MONTH DATA.
      MS(2) = 28
      I = IY/4*4                            ;LEAP YEARS.
      IF (I.EQ.IY) MS(2) = 29
      I = IY/100*100
      IF (I.EQ.IY) MS(2) = 28
      I = IY/400*400
      IF (I.EQ.IY) MS(2) = 29
      MS(3) = 31
      MS(4) = 30
      MS(5) = 31
      MS(6) = 30
      MS(7) = 31
      MS(8) = 31
      MS(9) = 30
      MS(10) = 31
      MS(11) = 30
      MS(12) = 31
      J = 0                                 ;SEARCH FOR MONTH.
      DO 100 I = 1,12
      M = I
      J = J + MS(I)
      IF (J.GE.JDA) GOTO 200
  100 CONTINUE
      M = 0                                 ;JDA IS TOO BIG.
      RETURN
  200 IF (J.GT.JDA) GOTO 300
      ID = MS(M)
      RETURN
  300 ID = JDA + MS(M) - J
      RETURN
      END
```

Eastern Region Computer Programs and Problems (Continued)

19 Verification of Asynchronous Transmissions. Lawrence Cedrone, March 1984. (PB84 189885)

20 AFOS Hurricane Plotter. Charles Little, May 1984. (PB84 199629)

21 WARN - A Warning Formatter. Gerald G. Rigdon, June 1984. (PB84 204551)

22 Plotting TDL Coastal Wind Forecasts, Paula Severe, June 1984 (Revised) (PB84-220789)

23 Severe Weather Statistics STADTS Decoder (SWX) and Plotter (SWY), Hugh M. Stone, June 1984. (PB84-213693)

24 WXR, Harold Opitz, August 1984. (PB84-23722)

25 FTASUM: Aviation Forecast Summaries, Matthew Peroutka, August 1984. (PB85-112977)

26 SAOSUM: A Short Summary of Observations. Matthew Peroutka, October 1984. (PB85-120384)

27 TRAJ - Single Station Trajectory Plot, Tom Niziol, December 1984. (PB85-135002)

28 VIDTEX, Gerald G. Rigdon, February 1985 (PB85-175669/AS)

29 ISENTROPIC PLOTTER, Charles D. Little, February 1985 (PB85-175651/AS)

NOAA SCIENTIFIC AND TECHNICAL PUBLICATIONS

The National Oceanic and Atmospheric Administration was established as part of the Department of Commerce on October 3, 1970. The mission responsibilities of NOAA are to assess the socioeconomic impact of natural and technological changes in the environment and to monitor and predict the state of the solid Earth, the oceans and their living resources, the atmosphere, and the space environment of the Earth.

The major components of NOAA regularly produce various types of scientific and technical information in the following kinds of publications:

PROFESSIONAL PAPERS — Important definitive research results, major techniques, and special investigations.

CONTRACT AND GRANT REPORTS — Reports prepared by contractors or grantees under NOAA sponsorship.

ATLAS — Presentation of analyzed data generally in the form of maps showing distribution of rainfall, chemical and physical conditions of oceans and atmosphere, distribution of fishes and marine mammals, ionospheric conditions, etc.

TECHNICAL SERVICE PUBLICATIONS — Reports containing data, observations, instructions, etc. A partial listing includes data serials; prediction and outlook periodicals; technical manuals, training papers, planning reports, and information serials; and miscellaneous technical publications.

TECHNICAL REPORTS — Journal quality with extensive details, mathematical developments, or data listings.

TECHNICAL MEMORANDUMS — Reports of preliminary, partial, or negative research or technology results, interim instructions, and the like.

Information on availability of NOAA publications can be obtained from:

ENVIRONMENTAL SCIENCE INFORMATION CENTER (D822)
ENVIRONMENTAL DATA AND INFORMATION SERVICE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE

6009 Executive Boulevard
Rockville, MD 20852