

UR1983-27

1/5

1983/27. A FORTRAN program for sorting areally distributed data to minimise plotting time (Revision 1).

R.G. Richardson

*Abstract*

This program sorts randomly distributed sample data into north-south strips and then orders the data within each strip to have increasing or decreasing north co-ordinates to minimise plotter pen movements.

THE PROGRAM

*SORTST (Appendix 1)*

The program reads the east co-ordinates of the sample points from a random access disk file and uses an indexed sort to order the data to have increasing east co-ordinates. The ordered data is divided into north-south strips of user-specified width. For each strip, the north co-ordinates of the sample points are read, sorted into increasing or decreasing order as appropriate, and the original sample data for that strip copied to the output file in the new order.

Control data is read from logical unit 5, the sample data is read from logical unit 4 (random access) and the sorted file is on logical unit 6.

Control data read from logical unit 5 is:

XMIN - the western edge of the area containing the samples (km)  
XNOM - the width of the north-south strips (km)

Data input from logical unit 4 is:

X, Y - format (10X, 2F9.1)  
- the east and north co-ordinate of the sample point in metres.

Data output to logical unit 6 is:

An exact copy of the first 80 characters of the corresponding record on logical unit 4.

[11 July 1983]

APPENDIX 1  
Program SORTST

```

$TITL SORTST.FTN - SORTING STATIONS INTO GEOGRAPHICAL ORDER
C A PROGRAM TO SORT AREALLY DISTRIBUTED DATA TO OPTIMISE PLOTTING THE DATA
C THE DATA IS SORTED INTO ASCENDING X-COORDINATE VALUES. N-S STRIPS ARE
C TAKEN AND THE STATIONS WITHIN EACH STRIP SORTED INTO ORDERED Y VALUES
C AND THEN OUTPUT IN THIS ORDERED FORM.
C INPUT DATA LU 4 RANDOM ACCESS
C OUTPUT DATA LU 6
C CONTROL DATA LU 5
C ASSUMES DATA IS ON A NOMINAL GRID AND READS THE STARTING GRID X VALUE
C AND THE NOMINAL X SPACING FROM LU 5
      INTEGER*2 INDEX(5000)
      REAL X(5000), ALINE(20)
      LOGICAL UP, IER
      OPEN(UNIT=4, ACCESS='DIRECT')
C OPEN THE INPUT FILE FOR RANDOM ACCESS
      OPEN(UNIT=5, FILE='CON:')
C OPEN THE CONSOLE FOR COMMANDS
      WRITE(5, 200)
      200 FORMAT(' ENTER XMIN AND XNOM IN KM. '// XMINNNNNN XNOMMMMM')
      READ(5, *) XMIN, XNOM
C HAVE STARTING VALUE AND NOMINAL GRID SPACING
      XMIN=XMIN*1000.0
      XNOM=XNOM*1000.0
C CONVERT TO METRES
      REWIND 4
C REWIND DATA FILE
      N=1
C COUNTER FOR NUMBER OF DATA VALUES
      10 READ(4, 100, END=11) X(N)
      100 FORMAT(10X, F9.1)
C STORE AWAY
      INDEX(N)=N
C FILL SORT INDEX ARRAY
      N=N+1
C INCREMENT N
      IF (N .EQ. 5001) STOP 888
C TOO MANY POINTS
      GOTO 10
C BACK ROUND
C
C NOW SORT INTO ASCENDING X VALUES
      11 N=N-1
C COUNT WAS PRE-INCREMENTED
      CALL BUBSTR(INDEX, X, 1, N)
C PERFORM THE SORT
C
C NOW DIVIDE INTO STRIPS XNOM WIDE AND SORT ON Y VALUES WITHIN THE STRIPS
      UP=.TRUE.
C USED TO OPTIMISE PEN MOVEMENT BY ORDERING OUTPUT IN INCREASING Y VALUE
C THEN IN DECREASING Y VALUE
      XMIN=XMIN-0.5*XNOM
C WANT TO CENTRE STRIPS ABOUT THE NOMINAL POSITION
      IBOT=1
      15 CONTINUE
      XUP=XMIN+XNOM
C EASTERN SIDE OF STRIP
      J=0
C COUNTER FOR NUMBER OF STATIONS IN STRIP

```

```

      DO 20 I=IBOT,N
      INDX=INDEX(I)
      IF (X<INDX) .GT. XUP) GOTO 22
C THIS IS THE FIRST STATION TO THE EAST OF THE STRIP
      J=J+1
C FIND THE DATA RECORD USING A RANDOM SEEK
      READ(4,101,REC=INDX) X<INDX>
      101 FORMAT(19X,F9.1)
C REPLACE THE X VALUE WHICH IS NOW REDUNDANT BY THE Y VALUE
      20 CONTINUE
      I=N+1
C ONLY GET TO HERE WHEN HAVE SCANNED ALL DATA ON THE FILE
      22 CONTINUE
      IF (J .EQ. 0) GOTO 24
C NO POINTS IN THIS STRIP
C
C IF POINTS IN THE STRIP ORDER THEM IN Y
      CALL SUBSTR<INDEX<IBOT>, X, 1, J>
C SORT
      JBOT=IBOT-1
      INC=+1
C PARAMETERS FOR OUTPUTTING IN ASCENDING Y
      IF <UP> GOTO 21
      JBOT=IBOT+J
      INC=-1
C SET PARAMETERS FOR SCANNING DOWN IN Y FOR OUTPUT
      21 CONTINUE
      DO 23 K=1, J
      JBOT=JBOT+INC
C FIND THE RECORD IN THE ORIGINAL FILE TO COPY THE COMPLETE
C RECORD TO THE OUTPUT FILE
      READ(4,102,REC=INDEX<JBOT>>) ALINE
      WRITE(6,102) ALINE
      102 FORMAT(20A4)
C COPY HERE
      23 CONTINUE
      UP=.NOT. UP
C CHANGE THE OUTPUTTING DIRECTION
      24 IBOT=I
C SET BASE FOR NEXT STRIP SEARCH
      IF (I .GT. N) GOTO 25
C BEEN THROUGH ALL STATIONS SO TIDY UP
      XMIN=XUP
C MOVE WEST EDGE OF STRIP OVER
      GOTO 15
C ROUND AGAIN
C
C TIDY UP
      25 CONTINUE
      CLOSE<UNIT=6>
      STOP
      END
      SUBROUTINE SUBSTR<IR, A, IBASE, N>
      REAL A<N>
      INTEGER*2 IR<N>
      LOGICAL NSWAP
      IF (N .LE. 1) RETURN
C NOTHING TO SORT
      NM1=N-1
      DO 30 J=IBASE, NM1

```

```
NSWAP=. TRUE.  
IRI=IR(1)  
DO 40 I=IBASE, NM1  
IP1=I+1  
IRIP1=IR(IP1)  
IF (A(IRI) .LE. A(IRIP1)) GOTO 40  
NSWAP=. FALSE.  
IR(I)=IRIP1  
IR(IP1)=IRI  
IRIP1=IRI  
40 IRI=IRIP1  
IF (NSWAP) RETURN  
30 CONTINUE  
RETURN  
END
```