1 SUMMARY

To **plot the part of a conic section that is inside a given triangle.** The conic section is defined by the equation $f(x,y) = \eta$, where η is the contour height and where $f(x,y)$ is a quadratic function that is specified by its value at the vertices and midpoints of the triangle sides.

A triangular region ABC defined by the coordinates of its three vertices A, B and C is given. The function values at the points A,B,C and P,Q,R are given where P,Q,R are the midpoints of the sides BC, CA and AB of the triangle. The quadratic function

$$f(x,y) = ax^2 + 2hxy + by^2 + 2gx + 2fy + c$$

is uniquely defined by the function values at the six points A,B,C,P,Q and R.

This routine plots the part of the contour line $f(x,y) = \eta$ that is inside or on the boundary of the triangular region. The contour line is drawn on the graph plotter through the Harwell GHOST graph plotting package. The user must initiate the use of the graph plotter and must define axes and scaling before calling OB13A. The subroutine uses the GHOST graphics package but communicates with GHOST through only one subroutine which makes it easily adapted for other graphics packages.

The method is described in Marlow and Powell, *A Fortran Subroutine for Plotting the Part of a Conic that is Inside a Given Triangle*, AERE R.8336 (1976).

ATTRIBUTES — **Version:** 1.0.0. **Types:** OB13A. **Calls:** FD05, SHVECS (a GHOST subroutine). **Original date:** March 1976. **Origin:** S.Marlow, Harwell.

2 HOW TO USE THE PACKAGE

2.1 The argument list and calling sequence

There is only a single precision version of this routine:

```
CALL OB13A(FD,XD,YD,ETA,DELT,NPT,XARC,YARC)
```

FD   is a REAL array of length 6, which must be set by the user so that FD(1), FD(2), FD(3),FD(4), FD(5), FD(6) contain the function values at the points A, B, C, P, Q and R respectively.

XD   is a REAL array of length 3, which must be set by the user so that XD(1), XD(2) and XD(3) contain the values of the abscissae at the points A,B and C respectively.

YD   is a REAL array of length 3, which must be set by the user so that YD(1), YD(2) and YD(3) contain the values of the ordinates at the points A, B and C respectively.

ETA   is a REAL variable which must be set by the user to the contour height required, η.

DELT is a REAL variable which must be set by the user to the value of the tolerance for the plotted contour. In practice the contour line is plotted as a sequence of straight line segments, and so will deviate from the smooth conic. If the given value of DELT is positive and not absurdly small then the straight line segments will not deviate from the smooth conic by more than the value DELT. For other values of DELT a default value will be calculated and used. If the user does not want to think about the accuracy, the value of DELT can be set to zero.

NPT   is an INTEGER is used in controlling the output of the contour to the graph plotter. It must initially be set by the user. It has the following values:

    $-1 \leq$ NPT ≤ 1 : Any previous contouring has been output to the graph plotter.

|NPT| ≥ 2 (Set by subroutine) There are |NPT| joins representing the contour awaiting output to the plotter.

NPT ≥ 0 (Set initially by the user) Any contour is output to the graph plotter before returning to the calling program. The final value of NPT is zero.

NPT 0 (Set initially by the user) Any contour calculated by the routine is not output on this call. NPT is set to a positive value which gives the number of joins waiting output.

    If OB13A is to be used only once, then set NPT=0 initially. If OB13 is used to draw a contour line through several triangles, set NPT=1 initially and then leave NPT alone until just before the call to the routine for the last triangle when the value of NPT is negated. Discontinuities in contour lines are dealt with automatically by a previous contour being output before a new branch of a contour line is approximated.

XARC is a REAL array of length 100, (see Section 3) which is set by the routine to contain the x coordinates of the joins (x_i, y_i) which is the straight line segment approximation to the contour. When this array becomes full, it and YARC is output to the graph plotter and so there are no restrictions on the use of the subroutine.

YARC is a REAL array of length 100, (see Section 3) which is set by the routine to contain the y coordinates of the joins (x_i, y_i) which is the straight line segment approximation to the contour. When this array becomes full, it and XARC is output to the graph plotter and so there are no restrictions on the use of the subroutine.

2.2 Use of labelled COMMON

The subroutine includes the statement

```
COMMON/OB13B/DD,IARC
```

where

DD   is a REAL variable defaulted to the value 1000.0. The subroutine ensures that the error of the piecewise linear approximation is at most the quantity

      max (DELT, largest side of triangle/DD)

IARC is an INTEGER, defaulted to the value of 100. It is set to the length of the REAL arrays XARC and YARC.

3 GENERAL INFORMATION

Use of common: See Section 2.2.

Workspace: Two real arrays of length 127 are used for workspace. They do not restrict the use of OB13 in any way. **Other Routines :** Subroutine PTJOIN is called, which is a part of the GHOST graph plotting package. This is easily replaced by an equivalent from a local package.

Input/output: None **Remark:** Whenever any output is made to the plotter it is done so via a call to subroutine OB13Z(XARC,YARC,NPT).

4 METHOD

    This is described in a Harwell report by S. Marlow and M.J.D. Powell, *A Fortran Subroutine for Plotting the Part of a Conic that is Inside a Given Triangle* (AERE R)8336).

5 EXAMPLE

Suppose we want to draw a contour for height ETA over several triangles (IST of them).

```
      CALL PAPER(1)                    switch on graph plotter
      CALL FRAME                       request new frame
      CALL PSPACE(.05,.95,.05,.95)     set up plotter space
```

```
      CALL CSPACE(0.,1.,0.,1.)              set up character space
      CALL MAP(0.,9.,0.,9.)                 set up mathematical space
      NPT=1                                 we want to continue contour across
      ETA=                                  triangles
      DELT=0.0                              subroutine will calculate suitable valu
      DO 1 I=1,IST                          loop for several triangles
      XD(1)=                                )
      XD(2)=                                )
      XD(3)=                                )
      YD(1)=                                ) set up coordinates of triangle and
      YD(3)=                                ) function values
      FD(1)=                                )
         ---                                )
         ---                                )
         ---                                )
      FD(6)=                                )
      IF(I.EQ.IST)NPT=-NPT                  on the last triangle we want to
                                            ensure all contours have been
                                            output
      CALL OB13A(FD,XD,YD,ETA,DELT,NPT,XARC,YARC)
    1 CONTINUE
      CALL GREND                            switch off graph plotter
```