
Subject: Re: Linking IDL with FORTRAN routines under UNIX: interpol.f
Posted by [thompson](#) on Tue, 24 Sep 1991 19:54:13 GMT
[View Forum Message](#) <> [Reply to Message](#)

This is the FORTRAN source code for the sample program on a Sun workstation.
Please note that I have not been able to use CALL_EXTERNAL with FORTRAN
subroutines on the SUN.

```
      SUBROUTINE INTERPOL(X_IN,Y_IN,N_IN,X_OUT,Y_OUT,N_OUT)
C+
C NAME:
C INTERPOL
C PURPOSE:
C Perform bilinear interpolation. Called using CALL_EXTERNAL from the
C IDL routine INTERPOL.
C CALLING SEQUENCE:
C CALL INTERPOL(X_IN,Y_IN,N_IN,X_OUT,Y_OUT,N_OUT)
C INPUT PARAMETERS:
C X_IN = Input array of X values.
C Y_IN = Input array of Y values.
C N_IN = Number of elements in X_IN, Y_IN.
C X_OUT = Array of points along the X axis to interpolate to.
C N_OUT = Number of points in X_OUT, Y_OUT.
C OUTPUT PARAMETERS:
C Y_OUT = Array of interpolated points.
C COMMON BLOCKS:
C None.
C SIDE EFFECTS:
C None.
C RESTRICTIONS:
C None.
C PROCEDURE:
C Straightforward.
C MODIFICATION HISTORY:
C William Thompson, August 1991.
C-
C
C      DIMENSION X_OUT(N_OUT),Y_OUT(N_OUT),X_IN(N_IN),Y_IN(N_IN)
C
C      Find the interval in the input arrays that corresponds to the point to be
C      interpolated to.
C
C      DO I = 1,N_OUT
C        J = 2
C        IF (X_IN(2) .GT. X_IN(1)) THEN
C          DO WHILE ((X_IN(J) .LT. X_OUT(I)) .AND. (J .LT. N_IN))
C            J = J + 1
C          ENDDO
```

```

ELSE
  DO WHILE ((X_IN(J) .GT. X_OUT(I)) .AND. (J .LT. N_IN))
    J = J + 1
  ENDDO
ENDIF
C
C Calculate the slope and intercept of the line connecting the two points
C defining the interval found above, and perform the interpolation.
C
  SLOPE = (Y_IN(J) - Y_IN(J-1))/(X_IN(J) - X_IN(J-1))
  Y_OUT(I) = Y_IN(J-1) + SLOPE*(X_OUT(I) - X_IN(J-1))
C
ENDDO
C
RETURN
END

```
