ADAS Subroutine xxsplf

```
SUBROUTINE XXSPLF( LSETX , LSETY , IOPT , FINTX ,
                        NIN , XIN , YIN
                        NOUT , XOUT , YOUT
    &
                            , DY
    &
                              , D1 , D2 , D3 ,
                        LINTRP
С
  ********** FORTRAN77 SUBROUTINE: XXSPLF ***************
С
С
                     TO INTERPOLATE/EXTRAPOLATE USING CUBIC SPLINES
С
 PURPOSE:
С
                     (IF IOPT < 0 NO EXTRAPOLATION TAKES PLACE = VALUES
С
С
                      SET TO ZERO). - LOGICAL ARRAY 'LINTRP()' SPECIFIES
                      WHETHER OUTPUT SPLINE IS INTERPOLATED '.TRUE.' OR
С
C
                      EXTRAPOLATED '.FALSE.'.
C
С
                      (AS FOR 'XXSPLN' EXCEPT 'LINTRP' ARGUMENT ADDED).
С
                      (AS FOR 'XXSPLE' EXCEPT WITH OPTION TO USE
С
                       PREVIOUSLY CALCULATED SPLINE DERIVATIVES)
С
С
 CALLING PROGRAMS: GENERAL USE
С
C SUBROUTINE:
С
С
 I/O : (L*4) LSETX = .TRUE. => SET UP SPLINE PARAMETERS RELATING
С
                                      TO 'XIN' AXIS.
С
                           .FALSE. => DO NOT SET UP SPLINE PARAMETERS
С
                                      RELATING TO 'XIN' AXIS.
С
                                      (I.E. THEY WERE SET IN A PREVIOUS
С
                                            CALL )
С
                           ( 'LSETX' IS ALWAYS RETURN AS '.FALSE.' ON
С
                             RETURN FROM THE SUBROUTINE ).
С
                           ** IMPORTANT: SEE NOTES BELOW ON 'LSETX' **
 I/O : (L*4) LSETY = .TRUE. => CALCULATE SPLINE DERIVATIVES
С
                                      RELATING TO 'YIN' AXIS.
С
С
                           .FALSE. => DO NOT SET UP SPLINE DERIVATIVES
С
                                      RELATING TO 'YIN' AXIS.
С
                                      (I.E. THEY WERE SET IN A PREVIOUS
С
                                            CALL )
С
                            ( 'LSETY' IS ALWAYS RETURN AS '.FALSE.' ON
С
                             RETURN FROM THE SUBROUTINE ).
                           ** IMPORTANT: SEE NOTES BELOW ON 'LSETY' **
С
С
  INPUT: (I \star 4) IOPT
                        = SPLINE END CONDITIONS/EXTRAPOLATION CONTROL
C
                           SWITCH - SEE NOTES BELOW
                           I.E. DEFINES THE BOUNDARY DERIVATIVES.
С
С
                                (VALID VALUES = 0, 1, 2, 3, 4)
С
                           IF IOPT < 0 THEN NO EXTRAPOLATION TAKES
С
                           - ANY VALUES REQUIRING EXTRAPOLATION WILL BE
С
                            SET TO ZERO (END CONDITIONS AS FOR IOPT=0)
C INPUT: (R*8) FINTX = INTERPOLATING X-COORDINATE TRANSFORMATION.
```

```
С
                           EXTERNAL FUNCTION (SEE ROUTINES BELOW)
С
C INPUT: (1*4) NIN
                       = NUMBER OF KNOTS
C INPUT: (R*8) XIN() = X-VALUES OF KNOTS
C INPUT: (R*8) YIN() = Y-VALUES OF KNOTS
С
C INPUT : (1 * 4) NOUT
                        = NUMBER OF OUTPUT VALUES TO BE INTERPOLATED
С
                          EXTRAPOLATED.
C INPUT: (R*8) XOUT() = X-VALUES AT WHICH INTERPOLATION/EXTRAPOLA-
С
                           TION REQUIRED
C OUTPUT: (R*8) YOUT()
                        = INTERPOLATED/EXTRAPOLATED Y-VALUES FOR
С
                           REQUESTED 'XOUT()' VALUES.
С
C I/O : (R*8)
                         = TRANSFORMED VALUES OF 'XIN()'. (ARRAY SIZE:
                 X()
С
                           NIN) REQUIRED INPUT IF LSETX IS .FALSE.
C I/O : (R*8) DY()
                        = DERIVATIVES AT INPUT KNOTS. REQUIRED INPUT
С
                           IF LSETY IS .FALSE.
C I/O : (R*8) Q() = SECOND DERIVATIVE FOR KNOT. REQUIRED INPUT
С
                           IF LSETX IS .FALSE. AND LSETY IS .TRUE.
C I/O : (R*8) D1()
                        = MULTIPLICATION FACTOR USED IN CALCULATING
С
                           'U()'. REQUIRED INPUT IF LSETX IS .FALSE.
С
                           AND LSETY IS .TRUE.
С
 I/O : (R*8) D2()
                        = MULTIPLICATION FACTOR USED IN CALCULATING
С
                           'U()'. REQUIRED INPUT IF LSETX IS .FALSE.
С
                           AND LSETY IS .TRUE.
С
 I/O : (R*8) D3()
                         = MULTIPLICATION FACTOR USED IN CALCULATING
                           'U()'. REQUIRED INPUT IF LSETX IS .FALSE.
С
С
                           AND LSETY IS .TRUE.
С
С
  OUTPUT: (L*4) LINTRP() = .TRUE. => 'YOUT()' VALUE INTERPOLATED.
                           .FALSE. => 'YOUT()' VALUE EXTRAPOLATED.
С
С
                           (ARRAY SIZE: NOUT)
С
С
          (1*4) NKNOTS = PARAMETER = MAXIMUM NUMBER OF KNOTS ALLOWED
С
          (1 \star 4) NIOPT = PARAMETER = MAXIMUM VALUE OF IOPT ALLOWED
С
С
          (I \star 4) I
                        = GENERAL ARRAY USE
                        = INDEX OF 'XOUT()' VALUE FOR INTERPOLATION/
С
           (I \star 4) K
                           EXTRAPOLATION.
С
С
          (I \star 4) NINO
                        = 'NIN' - 1
С
           (1 \star 4) INTER = INDEX OF CLOSEST/NEXT HIGHEST VALUE OF
С
                           'XIN()' TO THE VALUE OF 'XOUT()' BEING
С
                           INTERPOLATED/EXTRAPOLATED. WHEN LOOPING
С
                           OVER MULTIPLE YOUT EVALUATIONS, THE INDEX
С
                           OF THE LAST EVALUATION IS USED AS THE
С
                           INITIAL GUESS FOR THE NEXT.
С
          (I \star 4) NOPT
                        = VALUE OF 'IOPT' USED IN CALCULATING END-
                           CONDITIONS FOR STORED 'X-VALUE' SPLINE
С
С
                           PARAMETERS. (NOTE: IF 'IOPT < 0', THEN
С
                           'NOPT = 0'.) - I.E. 'NOPT = MAX( 0, IOPT )'.
С
С
           (R*8) XK
                        = VALUE OF 'XOUT(K)' BEING INTERPOLATED/
```

EXTRAPOLATED

С

C	C	С	(R	R*8)	XKK	=	TRANSFORMED VALUE OF 'XOUT(K)' BEING
SIDE OF CURRENT KNOT. C (R*8) T2 = (CURRENT KNOT POSITION TO NEXT HIGHEST KNOT POSITION) DIVIDED BY 'TI' C (R*8) T3 = (CURRENT KNOT POSITION TO NEXT LOWEST KNOT POSITION) DIVIDED BY 'TI' C (R*8) T4 = INTERPOLATION PATCOR FOR CURRENT KNOT POSITION) DIVIDED BY 'TI' C (R*8) DL1 = (REQUESTED 'XOUT()' VALUE TO NEXT HIGHEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OFF'. C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'YU(NIN)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'YU(NIN)' : FUNCTION OF 'NOPT' C (R*8) HUNAL() = VALUE OF 'YU(NIN)' : FUNCTION OF 'NOPT' C (R*8) HINTER = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' R	C	С					INTERPOLATED/EXTRAPOLATED.
C	C		(R	₹*8)	T1	=	
C (R*8) T3 = (CURRENT KNOT POSITION TO NEXT LOWEST KNOT POSITION DIVIDED BY 'T1' C (R*8) DL1 = (REQUESTED 'XOUT()' VALUE TO NEXT HIGHEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST CONTROL OF	C						
C	C		(R	₹*8)	T2	=	·
C (R*8) T4 = INTERPOLATION FACTOR FOR CURRENT KNOT C (R*8) DL1 = (REQUESTED 'XOUT()' VALUE TO NEXT HIGHEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OF'. C C (R*8) DVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) DVAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) DVAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) DELY() = SEPARATION, A	C				_ 0		
C	C (R*8) T4 = INTERPOLATION FACTOR FOR CURRENT KNOT C (R*8) DL1 = (REQUESTED 'XOUT()' VALUE TO NEXT HIGHEST C XNOT POSITION) DIVIDED BY SEPARATION OF C KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C XNOT POSITION) DIVIDED BY SEPARATION OF C KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OFF'. C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FI		(R	₹*8)	Т3	=	
C (R*8) DL1 = (REQUESTED 'XOUT()' VALUE TO NEXT HIGHEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D4VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D4VAL() = VALUE OF 'YOUNN' : FUNCTION OF 'NOPT' C (R*8) D4VAL() = VALUE OF 'YOUNN' : FUNCTION OF 'NOPT' C (R*8) HONE OF 'YOUTH ON THE FOR CHARLES OF CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (L*4) LUVAL() = .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C (ALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE OF CALLS THEN THE VALUE OF 'LSETX' C (ALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE ON THE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C (ALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE ON THE SETS IT TO .FALSE. FOR YOU.	C (R*8) DL1 = (REQUESTED 'XOUT()' VALUE TO NEXT HIGHEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS SITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS SITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OFF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) BJVAL() = VALUE OF 'O3(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VOLUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VOLUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) H() = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C FUNCTION OF 'NOPT' C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FRAISE. FOR YOU. IF THE VALUE OF 'LETY' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LETY' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LETY' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LETY' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LETY' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LETY' C INTITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						•
C KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. .FALSE. => 'EXTRAPOLATION SWITCHED ON'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) B3VAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE OF CHANGE OF SLOPE AT FINAL POIN CONTAINS (C) (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (DATE OF CHANGE OF SLOPE AT FINAL POIN CONTAINS (C) (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTING C C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) DL2 = (REQUESTED 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT SEITHER SIDE OF 'XOUT(K)'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OFF'. C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) WAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) WAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) HOT SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C FUNCTION OF 'NOPT' C (SEET ST TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT IS CHANGED EBSTEN CALLS THEN THE VALUE OF 'LSETX' C NO ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXHIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXHIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN CHANGE OF 'NOPT' C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING CHANGED BETWEEN CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C SCHANGED BETWEEN CALLS	C (R+8) DL2 = (REQUESTED 'XOUT(K)'. C (R+8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R+8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST YAUT(K)'. C (R+8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL2'. C (L+4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'.		(R	₹*8)	DL1	=	, =
C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST COMEST C	C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST COMEST C						
KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED ON'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) MYAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C FUNCTION OF 'NOPT' C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBGOUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C NOTEY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C KNOT POSITION) DIVIDED BY SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'O13(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'NOPT' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		(D		DIO		
C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) MAGAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINE SUALUE OF 'UVAL()' REFERS	C (R*8) DL2 = (REQUESTED 'XOUT(K)'. C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		(R	(*8)	DL2	=	
C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXHIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXHIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT'	C (R*8) DL2 = (REQUESTED 'XOUT()' VALUE TO NEXT LOWEST C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF 'XOUT(K)' * 'DL1' * 'DL2'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF	C (R*8) DL3 = SEPARATION OF KNOTS EITHER SIDE OF		(5)		D.T.O		
C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. C .FALSE. => 'EXTRAPOLATION SWITCHED OF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) WVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (L*4) LUVAL() = .TRUE. SLOPE C (L	C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. C						
C C C C C C C C C C C C C	C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. .FALSE. => 'EXTRAPOLATION SWITCHED ON'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C FUNCTION OF 'NOPT' C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		(R	(*8)	DL3	=	
C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'. C .FALSE. => 'EXTRAPOLATION SWITCHED ON'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) WVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE COALLS THEN THE VALUE OF 'NOPT' C IS CHANCED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (L*4) LEXTRP = .TRUE. => 'EXTRAPOLATION SWITCHED ON'FALSE. => 'EXTRAPOLATION SWITCHED OFF'. C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) WAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) DELY() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C FUNCTION OF 'NOPT' C (SOLUTION OF 'NOPT' C (SOLUTION OF 'NOPT' C (SALUS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C (SON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C (NOTES INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C (NO THE FIRST CALL OF THIS SUBROUTINE.						'XOUT(K)' * 'DL1' * 'DL2'.
C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) OVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C FUNCTION OF 'NOPT' C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C (SCHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C (NO THE FIRST CALL OF THIS SUBROUTINE.		/ T	. 43	THUMBD		TOUR A LEVERAROI ATTON ON THE ONLY
C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL POIN SLOPE (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING COLLISHED ON ONT RESET 'LSETX' - THIS SUBROUTINE COLLISHED ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C C C C C C C C C C C C C C C C C C C		(1	·* 4)	LEXTRP	=	
C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN .FALSE.=> VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) QVAL() = VALUE OF 'Q(1)' : FUNCTION OF 'NOPT' C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C (NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						.FALSE. => 'EXTRAPOLATION SWITCHED OFF'.
C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) WVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN .FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL POIN SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) D2VAL() = VALUE OF 'D2(1)' : FUNCTION OF 'NOPT' C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U (NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C FUNCTION OF 'NOPT' C SLOPE C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		(D	0 \	OT77 T ()		WALLE OF LO(1) L. FUNCTION OF LNORTH
C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U (NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN .FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL POIN SLOPE C FUNCTION OF 'NOPT' C SLOPE C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) D3VAL() = VALUE OF 'D3(1)' : FUNCTION OF 'NOPT' C (R*8) UVAL() = VALUE OF 'U(NIN)' : FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C .FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C FUNCTION OF 'NOPT' C SLOPE C TOTOS: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'NOPT' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C (R*8) UVAL() = VALUE OF 'U(NIN)': FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) UVAL() = VALUE OF 'U(NIN)': FUNCTION OF 'NOPT' C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C SCALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO FINAL POIN SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) AGRL() = POLYNOMIAL CONSTANTS FOR CUBIC SPLINE FOR GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C FUNCTION OF 'NOPT' C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C GIVEN 'XOUT(K)' VALUE. C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT SLOPE C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) H() = SEPARATION, ALONG X-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C		(R	(*0)	AGRL()		
C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN SLOPE C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		(1)	0.01	ш/\		
C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN .FALSE.=> VALUE OF 'UVAL()' REFERS TO RATE .FALSE.=> VALUE OF 'UVAL()' R	C (R*8) HINTER = SEPARATION, ALONG X-AXIS, IN INTERVAL FOR INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C		(1)	(*0)	Π()		
C INTERPOLATION C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN .FALSE. => VALUE OF 'UVAL()' REFERS TO FINE SLOPE C SLOPE C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C .FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		<i>(</i> D) .i. Q \	птильр		
C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEX HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POIN FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL C SLOPE C SLOPE C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) U() = TEMPORARY STORAGE OF DECOMPOSED FACTORS C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.		(1)	(^ ()	111111111	_	
C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT OF SLOPE C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'NOPT' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (R*8) DELY() = SEPARATION, ALONG Y-AXIS, OF KNOT FROM NEXT HIGHEST KNOT. C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C .FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C .SLOPE C .FUNCTION OF 'NOPT' C .VOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C .ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C .ON THE FIRST CALL OF THIS SUBROUTINE.		(R	2 + 8)	II ()	=	
C HIGHEST KNOT. C (L*4) LUVAL()= .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT OF CHANGE OF SLOPE AT FINAL POINT OF SLOPE C SLOPE C FUNCTION OF 'NOPT' C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CHANGE OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (L*4) LUVAL()= .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. C .FALSE.=> VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C .FUNCTION OF 'NOPT' C .FUNCTION OF 'NOPT' C .USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C .CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C .ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C .SETS IT TO .FINAL AS BEING EQUAL TO .TRUE. NOPT IS C .SETS IT TO .FINAL AS BEING EQUAL TO .TRUE. NOPT IS C .SETS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C .ON THE FIRST CALL OF THIS SUBROUTINE.						
C C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT E SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE FUNCTION OF 'NOPT' NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. ON THE FIRST CALL OF THIS SUBROUTINE.		(1)	(^0)	рппт ()		
C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT OF CHANGE OF 'UVAL()' REFERS TO FINAL SLOPE C SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C (L*4) LUVAL() = .TRUE. => VALUE OF 'UVAL()' REFERS TO RATE OF CHANGE OF SLOPE AT FINAL POINT. FALSE. => VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						mionibol invol.
C OF CHANGE OF SLOPE AT FINAL POINT C .FALSE.=> VALUE OF 'UVAL()' REFERS TO FINAL C SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	OF CHANGE OF SLOPE AT FINAL POINT. C .FALSE.=> VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C .FUNCTION OF 'NOPT' C .FUNCTION OF 'NOPT' C . USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C .ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C .ON THE FIRST CALL OF THIS SUBROUTINE. C		(T	·* 4)	T.UVAT.()	=	.TRUE. => VALUE OF 'UVAL()' REFERS TO RATE
C .FALSE.=> VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C .FUNCTION OF 'NOPT' C C .FUNCTION OF 'NOPT' C .VOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C .CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'LSETX' C .ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C .FALSE.=> VALUE OF 'UVAL()' REFERS TO FINAL SLOPE C .FUNCTION OF 'NOPT' C C .NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C .USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C .CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C .SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C .IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C .ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C .INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C .ON THE FIRST CALL OF THIS SUBROUTINE.		(-	/	20112()		• •
C SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C SLOPE C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C FUNCTION OF 'NOPT' C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C FUNCTION OF 'NOPT' C C C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						
C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C NOTES: 'LSETX': SET TO .TRUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE.						FUNCTION OF 'NOPT'
C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. ON THE FIRST CALL OF THIS SUBROUTINE.						
C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF C CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C USED. IF THE 'XIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETX' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. ON THE FIRST CALL OF THIS SUBROUTINE.	С	NOTES: 'LSE	ETX':	SET TO	.TF	RUE. ON ENTRY IF A NEW 'XIN' ARRAY IS BEING
C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C SETS IT TO .FALSE. FOR YOU. IF THE VALUE OF 'NOPT' C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE. C	С					
C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C IS CHANGED BETWEEN CALLS THEN THE VALUE OF 'LSETX' C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE. C						
C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS	C ON ENTRY IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE. C	С			SETS IT	TC	.FALSE. FOR YOU. IF THE VALUE OF 'NOPT'
	C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE. C ON THE FIRST CALL OF THIS SUBROUTINE. C	С			IS CHAN	GEL	BETWEEN CALLS THEN THE VALUE OF 'LSETX'
C INITIALISED TO -1 SO THAT LSETX WILL BE SET .TRUE.	C ON THE FIRST CALL OF THIS SUBROUTINE.	С			ON ENT	RY	IS TAKEN AS BEING EQUAL TO .TRUE. NOPT IS
	C	С			INITIAL	ISE	CD TO -1 SO THAT LSETX WILL BE SET .TRUE.
C ON THE FIRST CALL OF THIS SUBROUTINE.		С			ON THE	FIF	RST CALL OF THIS SUBROUTINE.
	C MUDDEDODE LICENTAL MEED ONLY DE CEE EO EDITE						
C THEREFORE 'I CETY' NEED ONLY BE CET TO TOLLE ON ENTE		С					'LSETX' NEED ONLY BE SET TO .TRUE. ON ENTRY
	C IF ANY ONE OF THE FOLLOWING VALUES HAS CHANGED:	С			IF ANY	ONE	OF THE FOLLOWING VALUES HAS CHANGED:

С

C C C

C C

C C C

С

C C C C

C C C

С

С

C C C

C C C

C C C

> C C

C C C

C C C

С

C C

С

C C C

С

'NIN' , 'FINTX' , 'XIN(I), I=1,NIN'

'LSETY': SET TO .TRUE. ON ENTRY IF A NEW 'YIN' ARRAY IS BEING USED. IF THE 'YIN' AXIS IS THE SAME FOR A NUMBER OF CALLS THEN DO NOT RESET 'LSETY' - THIS SUBROUTINE SETS IT TO .FALSE. FOR YOU. IF LSETX IS .TRUE., EITHER ON ENTRY OR BECAUSE THE ROUTINE RESETS IT (SEE ABOVE) THEN LSETY IS ALSO SET TO .TRUE.

THEREFORE 'LSETY' NEED ONLY BE SET TO .TRUE. ON ENTRY IF YIN HAS CHANGED WHILE THE 'X' VALUES HAVE NOT.

CARE: VARIABLES MUST BE USED FOR 'LSETX' AND 'LSETY',
A CONSTANT, I.E. .TRUE., CANNOT BE DIRECTLY
TYPED AS AN ARGUMENT BECAUSE IT WILL BE CHANGED
TO .FALSE. ON RETURN.

SPLINE END CONDITIONS AND EXTRAPOLATION DEPEND ON 'IOPT' AS FOLLOWS:

	IOPT		NOPT		DV (1)	DDY(1)		DV (N)	DDV (NI)		XTRAP'	— — NT I
 -	10P1	 _ -	NOP I	 _ _	DY(1)	(ד) בעע	 	DY (N)	DDY(N)	_ I _ _ I _	AIRAP	— I IN I
ı	< 0	i	0	1	_	0.0		_	0.0	ı	NO	
i	0	i	0	i	_	0.0	i	_	0.0	i	YES	i
	1		1		_	0.0		-1.5	_		YES	
	2		2		0.0	_		1.0	_		YES	
	3		3		-0.5	_		-1.5	_		YES	
	4		4		0.0	_		_	0.0		YES	
-	5		5		-4.5	-		-1.5	_		YES	
	6		6		+0.5	_	- 1	_	0.0		YES	
	7		7		-3.5	_		_	0.0		YES	

NB. OPTIONS TO BE EXTENDED FOR POWER AND CX APPLICATION

IF (IOPT.LT.0) - NO EXTRAPOLATION TAKES PLACE VALUES SET TO ZERO (CARE IF LOG OF OUTPUT IS NEEDED).

IF (IOPT.GT.7) PROGRAM STOPS

THIS SUBROUTINE IS AN AMENDED AND STRUCTURED VERSION OF THE SUBROUTINE 'ESPLINE' WRITTEN BY H.P. SUMMERS, JET 26TH OCTOBER 1989. IT REMOVES THE COMMON BLOCK /IONSPL/, THE SWITCHES 'ISW & ISW2' AND ALSO THE CASE FOR THE INTERPOLATION OF CHARGE STATE VALUES. IT INTRODUCES THE FEATURE THAT AN ARRAY OF INPUT 'X-VALUES' CAN BE INTERPOLATED/EXTRAPOLATED IN ONE CALL.

C ROUTINES:

ROUTINE SOURCE BRIEF DESCRIPTION

```
С
С
          FINTX
                            EXTERNAL REAL*8 FUNCTION, USED TO
С
                             TRANSFORM X-COORDINATES.
С
          XXHUNT
                            SEARCH ROUTINE FOR FINDING INTERVAL
С
                             CONTAINING A PRESCRIBED VALUE IN A
С
                             MONOTONIC VECTOR. INITIAL GUESSES ARE
С
                              USED TO SPEED THE SEARCH.
С
С
C AUTHOR: LORNE D. HORTON (IPP GARCHING)
С
          L5.213
С
          IPP EXT. 1635
C DATE: 18/03/03
C-----
C Notes: AS FOR 'XXSPLE' BUT WITH 'LSETY' ADDED TO ALLOW
C
        EXTERNAL SAVING OF SPLINE COEFFICIENTS (FOR
         EXAMPLE, WHEN IT IS NECESSARY TO HOLD MORE THAN
С
С
         ONE SPLINE RESULT AT A TIME). IN ADDDITION, A
С
         HUNT ALGORITHM 'XXHUNT' FOR SPEEDING EVALUATION
С
        HAS BEEN ADDED.
С
С
C XXSPLE COMMENTS
С
C AUTHOR: PAUL E. BRIDEN (TESSELLA SUPPORT SERVICES PLC)
          K1/0/81
С
          JET EXT. 4569
С
C
         14/01/91 - ADAS91: AS FOR 'XXSPLN' BUT WITH 'LINTRP()' ADDED
C DATE:
C VERSION: 1.2
С
C MODIFIED: LORNE HORTON (JET) DATE: 25/10/97
          - ADDED IOPT CHOICES 5, 6 AND 7
C
C VERSION: 1.3
С
C MODIFIED: Martin O'Mullane (JET) DATE: 2/6/99
С
           - SAVE nin0 and inter variables also. All compilers, ie
С
            especially g77, do not automatically save (or initialise
С
             variables to zero).
C
C VERSION : 1.1
C DATE : 18-03-2003
C MODIFIED : Lorne Horton
С
            - First version.
С
C VERSION : 1.2
C DATE : 10-04-2007
C MODIFIED : Allan Whiteford
```

С	- Modif	ied documentati	on as part o	f automated	
С	subroutine document	ation preparati	on.		
C					
C					
С					
C					
	INTEGER	IOPT,	NIN,	NOUT	
	LOGICAL	LINTRP (NOUT)	,	LSETX,	LSETY
	REAL * 8	D1(NIN),	D2(NIN),	D3(NIN),	DY(NIN)
	REAL*8	Q(NIN),	X(NIN),	XIN(NIN)	
	REAL * 8	XOUT (NOUT),	YIN(NIN),	YOUT (NOUT)	