Science & Technology Facilities Council

TB15

PACKAGE SPECIFICATION | HSL ARCHIVE

1 SUMMARY

This subroutine finds **a periodic cubic spline $S(x)$ that interpolates n function values.** The user supplies the function values $f_1, f_2, \ldots f_n$ (with $f_1=f_n$) at points $x_1<x_2<\ldots<x_n$ (which need not be equally spaced) and the subroutine finds derivative values $S'(x_i)$ at the points x_i. $S(x)$ is cubic on each interval (x_i, x_{i+1}), i=1,2, . . . n–1, is continuous with two continuous derivatives at the points x_i, i=2,3, . . . n–1 and $S'(x_1) = S'(x_n)$, $S''(x_1) = S''(x_n)$. The subroutine `TG01` can be used to find values of $S(x)$.

ATTRIBUTES — **Version:** 1.0.0. **Types:** `TB15A, TB15AD.` **Calls:** None. **Original date:** April 1985. 72 lines. **Origin:** J.K.Reid, Harwell. **Remark:** This is a slightly rewritten version of `TB05` and supersedes it.

2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version

```
CALL TB15A(N,X,F,D,WORK,LP)
```

The double precision version.

```
CALL TB15AD(N,X,F,D,WORK,LP)
```

`N` is an `INTEGER` variable which must be set by the user to n, the number of data points. It is not altered. **Restriction:** $n\geq4$.

`X` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length at least n which the user must set to the points x_i, i=1,2, . . . n. It is not altered. **Restriction:** $x_1<x_2<\ldots<x_n$.

`F` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length at least n which the user must set to the function values $f_1, f_2, \ldots f_n$. It is not altered. **Restriction:** $f_1=f_n$.

`D` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length at least n which need not be set by the user. The subroutine puts the values of the first derivatives of the spline $S(x)$ at the points x_i in `D(i)`, i=1,2, . . . n.

`WORK` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length at least 3n which is used as a workspace. `W(1)` is set to zero on a successful return and to a nonzero value on a failure. The subroutine prints a diagnostic message when n<4, a point x_i is out of order (see `X`) or $f_1\neq f_n$ (see `F`).`W(1)` is set to 1,2,3 respectively.

`LP` is an `INTEGER` variable which must be set by the user to a unit number for printing or to a non-positive number if printing is to be suppressed.

3 GENERAL INFORMATION

Use of common: None.

Other routines called directly: None.

Input/output: Output is under the control of argument `LP`.

Restrictions: $n\geq4$, $x_1<x_2<\ldots<x_n$, $f_1=f_n$

4 METHOD

The continuity of the second derivative $S''(x)$ at x_i, i=2,3, . . .n−1, can be expressed as a linear equation involving $S'(x_{i-1})$, $S'(x_i)$ and $S'(x_{i+1})$. Similarly the continuity of $S''(x)$ at x_1 can be expressed as a linear equation involving $S'(x_{n-1})$, $S'(x_1)$ and $S'(x_2)$. Thus the unknown derivatives $S'(x_i)$, i=1,2, . . . n−1, satisfy a set of n−1 linear equations whose matrix is tridiagonal except for entries in positions (1,n−1) and (n−1,1). This set of equations is set up and solved by Gaussian elimination. No interchanges are needed because the matrix is diagonally dominant.

5 EXAMPLE OF USE

As a very simple example the following code finds a periodic cubic spline that interpolates the function values for the given data.

```
      DOUBLE PRECISION X(9),F(9),D(9),W(27)
      DATA X/0.0000D0,2.0000D0,4.0000D0,6.0000D0,8.0000D0,
     *       1.0000D1,1.2000D1,1.4000D1,1.6000D1/
      DATA F/0.0000D0,1.0000D0,2.0000D0,4.0000D0,5.0000D0,
     *       4.0000D0,3.0000D0,2.0000D0,0.0000D0/
      DATA N/9/
      DATA LP/6/
      CALL TB05AD(N,X,F,D,W,LP)
      WRITE(6,10)
10    FORMAT(10X,' X(I)',10X,' F(I)',10X,' D(I)')
      WRITE(6,20)(X(I),F(I),D(I),I=1,N)
20    FORMAT(/(3F16.4))
      STOP
      END
```

This produces the following output

```
  X(I)            F(I)            D(I)
  0.0000          0.0000         -0.2946
  2.0000          1.0000          0.6429
  4.0000          2.0000          0.7232
  6.0000          4.0000          0.9643
  8.0000          5.0000         -0.0804
 10.0000          4.0000         -0.6429
 12.0000          3.0000         -0.3482
 14.0000          2.0000         -0.9643
 16.0000          0.0000         -0.2946
```