1 SUMMARY

To interpolate the value of an even function \( f(x) \), i.e. such that \( f(-x) = f(x) \), given \( n \) function values \( f_i \) at points \( x_i, i = 1, 2, \ldots, n \) not necessarily equally spaced.

A polynomial \( P(x) \) of degree \( 2(n-1) \) is constructed such that \( P(x_i) = f_i \) and \( P(-x_i) = f_i, i = 1, n \) and based on the Lagrange interpolation formula. The coefficients of \( P(x) \) are not computed.


2 HOW TO USE THE PACKAGE

2.1 Argument list and calling sequence

The single precision version

CALL TB01A(X,F,XVAL,FVAL,N)

The double precision version

CALL TB01AD(X,F,XVAL,FVAL,N)

\( X \) is a REAL (DOUBLE PRECISION in the D version) array of length at least \( n \) which the user must set to contain the values of the points \( x_i, i = 1, 2, \ldots, n \). It is not altered by the subroutine. Restriction: the moduli of the points must be distinct, i.e. \( |x_i| \neq |x_j| \) for all \( i \neq j \).

\( F \) is a REAL (DOUBLE PRECISION in the D version) array of length at least \( n \) which the user must set to contain the function values \( f_i \) at the points \( x_i, i = 1, 2, \ldots, n \). The subroutine will assume that \( f(-x_i) = f(x_i) \). \( F \) is not altered by the subroutine.

\( XVAL \) is a REAL (DOUBLE PRECISION in the D version) variable which the user must set to the value of \( x \) for which the interpolated value of \( f(x) \) is required. \( XVAL \) is not altered by the subroutine.

\( FVAL \) is a REAL (DOUBLE PRECISION in the D version) variable which is set by the subroutine to the interpolated value of \( f(x) \) at the point given in \( XVAL \). It need not be set by the user.

\( N \) is an INTEGER variable which must be set by the user to \( n \), the number of function values passed in the array \( F \). It is not altered by the subroutine.

3 GENERAL INFORMATION

Use of common: none.

Workspace: none.

Other routines called directly: none.

Input/output: none.

Restrictions: \( |x_i| \neq |x_j| \) for all \( i \neq j \).
4 METHOD

The interpolation is based on the $2(n-1)$th degree even polynomial $P(x)$ which is equal to $f_i$ at the $2n$ points $x_i$ and $x_j$, $i=1, 2, ..., n$. A Lagrange interpolation formula is used and the number of operations per call is of order $n^2$. If many interpolations are required it will be more efficient to use another library subroutine to derive $P(x)$ explicitly since each interpolation value will then cost only of order $n$ operations.