1 SUMMARY

To interpolate the value of a function given \( n \) function values \( f_i \) at points \( x_i, i=1, 2, \ldots, n \), not necessarily equally spaced.

The interpolation is based on the \((n-1)\)th degree polynomial which passes through the \( n \) points obtained by the Lagrange interpolation formula. The coefficients of the polynomial are not computed.


2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version

\[
\text{CALL TB02A}(X,F,XVAL,FVAL,N)
\]

The double precision version

\[
\text{CALL TB02AD}(X,F,XVAL,FVAL,N)
\]

\( X \) is a REAL (DOUBLE PRECISION in the D version) array which must be set by the user to contain the values of the points \( x_i, i=1, 2, \ldots, n \). It is not altered by the subroutine. **Restriction:** all the points \( x_i, i=1, 2, \ldots, n \) must be different.

\( F \) is a REAL (DOUBLE PRECISION in the D version) array which must be set by the user to contain the values \( f_i, i=1, 2, \ldots, n \), of the tabulated function. It is not altered by the subroutine.

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

\( FVAL \) is a REAL (DOUBLE PRECISION in the D version) variable which will be set by the subroutine to the interpolated value of \( f(x) \) at the point given in \( XVAL \).

\( N \) 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: The \( x_i \) must be distinct.

4 METHOD

This subroutine evaluates the interpolated value from first principles at each call, consequently the number of operations is of the order of \( n^2 \). This is inefficient if many interpolations are required because it is then better to evaluate the explicit coefficients of the \((n-1)\)th order polynomial, so that the number of operations for each interpolation is of order \( n \).