Author(s) : F. James
Submitter :
Language : Fortran

Library: KERNLIB
Submitted: 05.09.1966
Revised: 18.11.1985

Polynomial Interpolation

Subroutine `POLINT` interpolates in a table of arguments a_j and function values $f_j = f(a_j)$, using an interpolating polynomial of specified degree $K-1$ which passes through K successive tabular points. The table arguments a_j need not be equidistant. Meaningful results can usually be obtained only for small values of K (typically less than 10).

Structure:

`SUBROUTINE` subprogram
User Entry Names: `POLINT`
Files Referenced: Printer
External References: `KERMTR` (N001), `ABEND` (Z035)

Usage:

```
CALL POLINT(F,A,K,X,R)
```

`F` (`REAL`) One-dimensional array. `F(j)` must be equal to the value at `A(j)` of the function to be interpolated, $(j = 1, 2, \ldots, \texttt{K})$.

`A` (`REAL`) One-dimensional array. `A(j)` must be equal to the table argument a_j, $(j = 1, 2, \ldots, \texttt{K})$.

`K` (`INTEGER`) `K-1` is the degree of the interpolating polynomial.

`X` (`REAL`) Argument at which the interpolating polynomial is to be evaluated.

`R` (`REAL`) On exit, `R` is set equal to the value at `X` of the polynomial passing through the points (a_j, f_j), $(j = 1, 2, \ldots, \texttt{K})$.

If `X` lies outside the range of the points `A(1)`, . . . , `A(K)`, the interpolation becomes an extrapolation, with consequent loss of accuracy.

Method:

Newton's divided difference formula is used.

Restrictions:

$2 \le \texttt{K} \le 20$. If $\texttt{K} > 20$, the interpolation is performed as if `K` had the value `20`. The original value of `K` is unchanged on exit.

Error handling:

Error `E100.1`: $\texttt{K} < 1$. A message is printed unless subroutine `KERSET` (N001) has been called.

Notes:

`POLINT` is intended for interpolation using *all* the tabulated points in the array `A`. To use only the tabulated points around the value of the argument `X`, use `DIVDIF` (E105).

•