Author(s) : G.A. Erskine
Submitter :
Language : Fortran

Library: KERNLIB
Submitted: 18.12.1979
Revised: 27.11.1984

Linear Equations, Matrix Inversion

Subroutine tEQN (where t = R, D or C as described below) solves the matrix equation

$$\mathbf{AX} = \mathbf{B}, \qquad (*)$$

which represents a system of N simultaneous linear equations with K right-hand sides:

$$\sum_{j=1}^{N} a_{ij}x_{jk} = b_{ik}, \qquad (i = 1,2,\ldots,N,\ k = 1,2,\ldots,K).$$

Subroutine tINV computes the inverse of a square matrix **A**. Subroutine tEQINV solves the system (*) and also computes the inverse of **A**, but is appreciably slower than tEQN.

If the determinant of **A** is also required, or if several systems of the form (*) are to be solved sequentially with the same coefficient matrix **A** but differing right-hand sides **B**, the subroutines in RFACT (F011) should be used.

Structure:

SUBROUTINE subprograms
User Entry Names: RINV, REQN, REQINV, DINV, DEQN, DEQINV, CINV, CEQN, CEQINV
Internal Entry Names: F010PR
Files Refeenced: Printer
External References: RFACT (F011), RFEQN (F011), RFINV (F011),
DFACT (F011), DFEQN (F011), DFINV (F011),
CFACT (F011), CFEQN (F011), CFINV (F011),
TMPRNT (F011), KERMTR (N001), ABEND (Z035)

Usage:

For t = R (type REAL), t = D (type DOUBLE PRECISION), t = C (type COMPLEX):

```
CALL tEQN  (N,A,IDIM,IR,IFAIL,K,B)
CALL tINV  (N,A,IDIM,IR,IFAIL)
CALL tEQINV(N,A,IDIM,IR,IFAIL,K,B)
```

N — (INTEGER) Order of the square matrix **A**.

A — (Type according to t) Two-dimensional array whose first dimension has the value IDIM.

IDIM — (INTEGER) First dimension of array A (and of array B if K > 1).

IR — (INTEGER) Array of at least N elements, required as working space.

IFAIL — (INTEGER) On exit, IFAIL will be set to −1 if **A** is found to be singular, and to 0 otherwise. (Singularity will often go undetected because of rounding errors during factorization even if the elements of **A** have integral values.)

K — (INTEGER) Number of columns of the matrices **B** and **X**.

B — (Type according to t) In general, a two-dimensional array whose first dimension has the value IDIM. B may be one-dimensional if K = 1.

These subroutines must be called with matrix **A** in array A and matrix **B** in array B. Then, provided the matrix **A** is non-singular, IFAIL will be set to 0 and arrays A and B will be set as follows:

tEQN The solution **X** replaces **B**. The matrix **A** is destroyed.

tINV The inverse $\mathbf{A}^{-1}$ of **A** replaces **A**.

tEQINV The solution **X** replaces **B**, and the inverse $\mathbf{A}^{-1}$ of **A** replaces **A**.

If the matrix **A** is singular, IFAIL will be set to -1. In this case the contents of A is unpredictable and the contents of B is unchanged.

Method:

Triangular factorization with row interchanges, implemented by in-line code if $\mathtt{N} \leq 3$ and by calls to library program RFACT (F011) if $\mathtt{N} > 3$. If $\mathtt{N} < 1$ or $\mathtt{IDIM} < \mathtt{N}$ or $\mathtt{K} < 1$, a message is printed and program execution is terminated by calling ABEND (Z035).

Examples:

Assume that the 10×10 matrix **A** and the 10×3 matrix **B** are stored according to the Fortran convention in arrays A and B respectively of a program containing declarations

```
DIMENSION IR(25)
DOUBLE PRECISION A(25,30),B(25,10)
```

To replace **B** by the 10×3 solution matrix **X** of the system of equations $\mathbf{AX} = \mathbf{B}$ and to replace **A** by $\mathbf{A}^{-1}$, with a jump to label 100 if **A** is singular:

```
CALL DEQINV (10,A,25,IR,IFAIL,3,B)
IF(IFAIL .NE. 0) GO TO 100
```

•