> **Warning:** Subroutine `MA10` performs functions which are adequately treated by routines in other standard subroutine libraries (for example, `LAPACK`). The use of this routine is not recommended, and it may be removed from future releases of this library.

1 SUMMARY

To **solve a system of m linear equations** in n unknowns with one or more right-hand sides.

$$\sum_{j=1}^{m} a_{ij}x_{jk} = b_{ik} \quad i=1,2,...,m \quad k=1,2,...,n$$

in the case when the coefficient $\mathbf{A}=\{a_{ij}\}_{m \times m}$ is **symmetric positive-definite.** It can also compute the inverse $\mathbf{A}^{-1}$.

Optionally, either the inverse operation or the solution operation can be omitted, or it will perform both functions at the same time. There is also a re-entry facility to obtain the inverse after a solutions only call. If the matrix $\mathbf{A}$ is found not to be positive definite an error return will be made and a diagnostic message printed.

Symmetric Cholesky decomposition is used with inner products accumulated double length.

ATTRIBUTES — Version: 1.0.0. **Remark:** Superseded by `MA22A`. **Types:** `MA10A`; `MA10AD`. **Original date:** May 1964. **Origin:** M.J.Hopper, Harwell.

2 HOW TO USE THE PACKAGE

2.1 The argument list and calling sequence

The single precision version:

        CALL MA10A(A,B,M,N,IERR,IOPT,IA,IB)

The double precision version:

        CALL MA10AD(A,B,M,N,IERR,IOPT,IA,IB)

A   is a two-dimensional `REAL` (`DOUBLE PRECISION` in the D version) array of dimensions at least m by m (first dimension specified in `IA`), in which the user passes the matrix $\mathbf{A}$. Only the elements `A(I,J)` `I≥J` of the lower triangle of `A` need be set on entry to the subroutine.

   The array is used as workspace and the upper triangle, `A(I,J)` `I≤J`, will be overwritten with the matrix factor $\mathbf{L}^T$, i.e. such $\mathbf{A}=\mathbf{L}\mathbf{L}^T$. If the inverse matrix has been requested and successfully computed it will be found complete in `A(I,J)` `I=1,M J=1,M` on return from the subroutine.

B   is a two-dimensional `REAL` (`DOUBLE PRECISION` in the D version) array of dimensions at least m by n (first dimension specified in `IB`), and must be set by the user to the n right hand sides b_{ik} $i=1,2,...,m$, $k=1,2,...,n$.

   If $n=1$ a single dimensioned array will suffice and `IB` is ignored whereas if $n \leq 0$ (requesting the inverse only) `B` and `IB` are ignored.

   On return from the subroutine after a successful call with `N>0` the solutions x_{jk} $j=1,2,...,m$ $k=1,2,...,n$ will be found in `B(J,K)` `J=1,M K=1,N`.

M   is an `INTEGER` variable and must be set by the user to m the number of equations. This argument is not altered by the subroutine. **Restriction:** $m>0$.

N   is an `INTEGER` variable and must be set by the user to n the number of right hand sides. Set $n \leq 0$ to suppress the solution calculation. This argument is not altered by the subroutine.

IERR   is an `INTEGER` variable which is set by the subroutine to signal the success of the calculation. It is returned set to zero if there have been no errors, and set to one if the matrix has been found not to be positive-definite. In this event the row number of the pivot that went negative is returned in the Common block (see §2.2), and a diagnostic message is printed.

IOPT   is an `INTEGER` variable and must be set by the user to select the operations to be performed. In combination with `N`, the number of right hand sides, these are as follows

> (a) inverse only, set `N≤0` and `IOPT>0`.
> (b) solve equations only, set `N>0` and `IOPT=0`.
> (c) inverse and solutions, set `N>0` and `IOPT>0`.
> (d) after calling a (b) entry the subroutine can be re-entered to complete the calculation of the inverse (which will have been partly done on the (b) call). To do this, set `IOPT<0` on re-entry, `N` is ignored.

IA   is an `INTEGER` variable and must be set by the user to the first dimension of the array `A`, e.g. if space for `A` is allocated by `DIMENSION A(100,100)`

set `IA` to 100. This argument is not altered by the subroutine.

IB   is an `INTEGER` variable and must be set by the user to the first dimension of the array `B`. It is ignored when `N≤1`. This argument is not altered by the subroutine.

2.2 The COMMON area

   The subroutine uses a Common block.

The single precision version:

        COMMON/MA10B/LP,IPIVOT

The double precision version:

        COMMON/MA10BD/ LP,IPIVOT

LP   is an `INTEGER` variable and specifies the Fortran stream number to be used to output diagnostic printing. It is defaulted to stream 6 (the line printer) but the user can change its value to any valid output stream number or set `LP≤0` to suppress diagnostic printing altogether.

IPIVOT is an `INTEGER` variable set by the subroutine to the row number of the first pivot to go negative in the event that the matrix is not positive-definite, i.e. when `IERR=1`. On a successful return `IPIVOT` is set to zero.

3 GENERAL INFORMATION

Use of Common: there is a Common block called `MA10B/BD`, see §2.2.

Workspace: none, the subroutine overwrites the array `A`.

Other subroutines: none.

Input/Output: in the event of errors a diagnostic message will be printed, see §2.2.

---

4 METHOD

Symmetric Choleski decomposition is used to find the lower triangle matrix $\mathbf{L}$ such that $\mathbf{A} = \mathbf{L}\mathbf{L}^T$. If there are equations to be solved the right hand sides are transformed to $\mathbf{L}^{-1}\mathbf{B}$ and the solutions computed by back substitution on the system $\mathbf{L}^T\mathbf{X} = \mathbf{L}^{-1}\mathbf{B}$.

If inverse is required $\mathbf{L}^{-1}$ is formed and $\mathbf{A}^{-1}$ computed from the equations $\mathbf{L}^T\mathbf{A}^{-1} = \mathbf{L}^{-1}$.

The method is described in 'Modern Computing Methods', N.P.L. (1957).