1 SUMMARY

This package of subroutines will factorize a matrix, solve corresponding systems of linear equations and update the factorization when a column of the matrix is altered, exploiting sparsity in all cases. Its primary application is likely to be for **handling linear programming bases.** It has four entries:

(a) LA15I/ID must be called for initialization.

(b) LA15A/AD factorizes a given matrix **A** of order *n*.

(c) LA15B/BD subsequently calculates $\mathbf{A}^{-1}\mathbf{b}$ or $\mathbf{A}^{-T}\mathbf{b}$ for a given vector **b**, using the factorized **A**.

(d) LA15C/CD modifies the factorization to correspond with the replacement of a column of the matrix by the vector **b** of a previous call of LA15B/BD which calculated $\mathbf{A}^{-1}\mathbf{b}$.

ATTRIBUTES — Version: 1.3.0. (9 April 2013) **Types:** Real (single, double). **Calls:** FD15, MC59. **Original date:** May 2001. **Remark:** The LA15 entries are threadsafe versions of LA05. **Origin:** J. K. Reid, Harwell.

2 HOW TO USE THE PACKAGE

2.1 Initialization argument lists and calling sequences

The LA15I/ID entry must be called to initialize the control arrays and private workspace prior to the first call to LA15A/AD.

The single precision version

```
CALL LA15I(ICNTL,CNTL,KEEP)
```

The double precision version

```
CALL  LA15ID(ICNTL,CNTL,KEEP)
```

ICNTL is an INTEGER array of length 3, see Section 2.5.

CNTL is a REAL (DOUBLE PRECISION in the D version) array of length 3, see Section 2.5.

KEEP is an INTEGER array of length 7 used by LA15 as private workspace. It must not be altered by the user.

2.2 Computation argument lists and calling sequences

The single precision version:

```
CALL LA15A(A,IND,NZ,IA,N,IP,IW,W,G,U,ICNTL,CNTL,KEEP)
CALL LA15B(A,IND,   IA,N,IP,IW,W,G,B,TRANS,ICNTL,KEEP)
CALL LA15C(A,IND,   IA,N,IP,IW,W,G,U,M,ICNTL,CNTL,KEEP)
```

The double precision version:

```
CALL LA15AD(A,IND,NZ,IA,N,IP,IW,W,G,U,ICNTL,CNTL,KEEP)
CALL LA15BD(A,IND,   IA,N,IP,IW,W,G,B,TRANS,ICNTL,KEEP)
CALL LA15CD(A,IND,   IA,N,IP,IW,W,G,U,M,ICNTL,CNTL,KEEP)
```

A is a REAL (DOUBLE PRECISION in the D version) array of length IA. On a call of LA15A/AD it must be set to contain, in any order, the entries of **A**. On return from LA15A/AD and on call of and return from any of the other LA15 routines, it contains the factors of the current matrix **A**. It is altered by LA15A/AD and LA15C/CD, and must not be altered by the user except prior to a call of LA15A/AD.

IND is an INTEGER array of dimensions (IA,2). On a call of LA15A/AD, IND(k,1) and IND(k,2) must be set to contain, respectively, the row and column number of the entry held in `A(k)` for `k`=1, 2,..., `NZ`. It is altered by `LA15A/AD` and `LA15C/CD`, and must not be altered by the user except prior to a call of `LA15A/AD`.

`NZ` is an `INTEGER` variable which must be set by the user to the number of entries in **A**. It is used by `LA15A/AD` only and is not altered by it. **Restriction:** `NZ` ≥ 0.

`IA` is an `INTEGER` variable which must be set by the user to indicate the size of arrays `A` and `IND`. Advice on the choice of the size is given in §2.3. It is not altered.

`N` is an `INTEGER` variable which must be set by the user to *n*, the order of **A**. It is not altered. **Restriction:** `N` ≥ 1.

`IP` is an `INTEGER` work array of length 2*n*. It must not be altered by the user except prior to a call of `LA15A/AD`.

`IW` is an `INTEGER` work array of length 8*n*, the first half of which must not be altered by the user except prior to a call of `LA15A/AD`. The second half is not used by `LA15B/BD` or `LA15C/CD`.

`W` is a `REAL` (`DOUBLE PRECISION` in the D version) work array of length at least *n*. It is used to transmit information about an incoming column between a call of `LA15B/BD` with `TRANS` = `.FALSE.` and a subsequent call of `LA15C/CD`, and therefore should not be altered between such calls.

`G` is a `REAL` (`DOUBLE PRECISION` in the D version) variable which is used to output information about the stability of the factorization and error conditions. After a successful call, `G` is positive and equal to the modulus of the largest element in any of the reduced matrices. This is explained further in §4. After an unsuccessful call, it is set negative (see §2.4 for details).

`U` is a `REAL` (`DOUBLE PRECISION` in the D version) variable which must be set by the user in the range $0 < U \leq 1$ to control the choice of pivots; if $U > 1$, it is reset to 1; if $U \leq 0$, it is reset to the relative floating-point accuracy. When searching for a pivot any element less than `U` times the largest element in its row is excluded. Thus decreasing `U` biases the algorithm towards maintaining sparsity at the expense of stability and vice-versa. The value 0.1 has been found satisfactory in test examples. It is used only by `LA15A/AD` and `LA15C/CD`.

`B` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length *n* used by `LA15B/BD` to input **b** and output $\mathbf{A}^{-1}\mathbf{b}$ (`TRANS` = `.FALSE.`) or $\mathbf{A}^{-T}\mathbf{b}$ (`TRANS` = `.TRUE.`). It is used by `LA15B/BD` only.

`TRANS` is a `LOGICAL` variable which must be set to `.FALSE.` if $\mathbf{A}^{-1}\mathbf{b}$ is required from `LA15B/BD` and to `.TRUE.` if $\mathbf{A}^{-T}\mathbf{b}$ is wanted. It is used only by `LA15B/BD` and is not altered by it.

`M` is an `INTEGER` variable which the user must set to the column number in **A** of the column to be replaced in a call of `LA15C/CD`. It is used by `LA15C/CD` only and is not altered by it. **Restriction:** 1 ≤ `M` ≤ `N`.

`ICNTL` is an `INTEGER` array of length 3 whose first element can optionally be set by the user, see Section 2.5.

`CNTL` is a `REAL` (`DOUBLE PRECISION` in the D version) array of length 3 whose first element can optionally be set by the user, see Section 2.5.

`KEEP` is an `INTEGER` array of length 7 used by `LA15` as private workspace and must not be altered by the user. `KEEP(1)`, `KEEP(2)` and `KEEP(3)` provide information about the use of the store and their use is described in §2.3.

2.3 Storage considerations

The matrix is factorized into a product **LU**, where **L** is a product of matrices $\mathbf{L}_i$ differing from the unit matrix **I** in only one element, and **U** is a permutation of an upper triangular matrix. Each matrix $\mathbf{L}_i$ is stored in one position of `A` and `IND`; the argument element `KEEP(1)` holds the number of such matrices. The number of entries in **U** is held in the argument element `KEEP(2)`. **U** is held in `A` and `IND`, as a file containing the matrix by rows and a file containing its structure by columns. These files will need occasional compression to release space used by altered rows and columns. This compression (actually performed by the subroutine `LA15E/ED`) will not add a significant overhead to the computational cost if it happens less often than, say, alternate calls of `LA15C/CD`. If it is required more than twenty times in a single call of `LA15C/CD`, this call is aborted and a diagnostic is printed. If such a call follows a long sequence of `LA15C/CD` calls, it can probably be corrected by a fresh `LA15A/AD` call. Both **L** and **U** are held in `A` and

IND, so IA (the size of A and IND) must exceed KEEP(1) + KEEP(2) (the number of entries in **L** and **U**) by a margin sufficient to avoid overfrequent compression of the files holding **U**. The adequacy of the length IA may be judged by monitoring the argument element KEEP(3), see §2.2, which accumulates the number of file compressions since the last call of LA15A/AD.

2.4 Error diagnostics

After an unsuccessful call a message is output on the line printer (unless suppressed or switched to another stream, see §2.5) and G is set negative. Possible values of G and corresponding messages are:

- −1 N IS NOT POSITIVE
- −3 ELEMENT ... IS IN ROW ... AND COLUMN ... (one of which is out of range)
- −4 THERE IS MORE THAN ONE ENTRY IN ROW ... AND COLUMN ...
- −5 THE MATRIX IS SINGULAR WITH RANK ......
- −6 SINGULAR MATRIX IS CREATED BY REPLACEMENT OF COL ...
- −7 IA IS TOO SMALL.
- −8 NZ IS NEGATIVE.
- −9 M HAS THE VALUE ... (which is out of range).

Any of the diagnostics except –6 may result from a call of LA15A/AD and diagnostic –6 or diagnostic –7 may result from a call of LA15C/CD. Also error returns (with G unchanged) may result from calling LA15B/BD or LA15C/CD with G negative, indicating a previous error return.

2.5 The control arrays

The ICNTL array argument which must be of length 3 can be used to pass optional integer control values to the routine.

ICNTL(1) specifies the unit number to be used to output error messages. A zero value will suppress output. It has a default value of 6, which is set by LA15I/ID.

ICNTL(2) and ICNTL(3) should not be altered and at present are not used by LA15.

The CNTL array argument which must be of length 3 can be used to pass optional real control values to the routine.

CNTL(1) can be used to specify that very small element values in the **LU** factorization of **A** are set to zero as explained in §4. It has a default value of zero, which is set by LA15I/ID.

CNTL(2) and CNTL(3) should not be altered and at present are not used by LA15.

3 GENERAL INFORMATION

Use of common: None.

Other routines called directly: Calls the private subroutine LA15E/ED and calls the HSL routines FD15A/AD and MC59A/AD.

Input/output: Output is under the control of argument ICNTL(1).

Restrictions: NZ ≥ 0, N ≥ 0, 1 ≤ M ≤ N.

4 METHOD

LA15A/AD factorizes **A** into a product **LU**, where **L** is a product of matrices $\mathbf{L}_i$ differing from the unit matrix **I** in only one element, and **U** is a permutation of an upper triangular matrix. It uses sparse matrix techniques similar to those of MA28.

Changing a column of **A** corresponds to changing a column of **U** so that it is no longer a permutation of a triangular matrix and further row operations and/or permutations are needed to restore it to this form.

To control stability, all pivots are chosen so that the multiples of a row that are added to another are always less than 1/`U` and stability is monitored by the parameter `G`, which is set to the modulus of the largest element in **A** or any of the upper triangular matrices to which it is reduced. If ε is the relative accuracy of the computation, then the solution obtained will have errors comparable with those of a perturbed system with matrix $\mathbf{A}+\delta\mathbf{A}$, elements of $\delta\mathbf{A}$ being less than a small multiple of εG. Any elements of **U** that are less than `CNTL(1)` are reset to zero; this has an effect comparable with that of making a perturbation to **A** whose elements have size about `CNTL(1)`. We recommend the user to reset this to a positive value if possible, because this will save most underflow interrupts and some storage.

An `LA15A/AD` call is normally followed by a long sequence of calls of `LA15B/BD` and `LA15C/CD`. The time taken by `LA15B/BD` will grow steadily as the number of entries in the factors of **A** grows and eventually it will be more economic to call `LA15A/AD` with the current matrix **A** and continue from this. A further call of `LA15A/AD` may also be needed because of instability; large values of `G` are an indication of trouble but a better test is to calculate $\mathbf{r} = \mathbf{A}\mathbf{x}-\mathbf{b}$ (or $\mathbf{A}^T\mathbf{x}-\mathbf{b}$), where **x** is the result of a `LA15B/BD` call, and compare r_i with $\sum_j |a_{ij}x_j|$ (or $\sum_j |a_{ji}x_j|$).

5 EXAMPLE OF USE

We illustrate the use of the package by reading and factorizing the matrix

$$\mathbf{A} = \begin{pmatrix} 2 & & & & \\ & 3 & 4 & & 6 \\ & & 1 & 5 & \\ & & 5 & & \\ & & & & 1 \end{pmatrix},$$

solving with the right-hand side $(1\ \ 4\ \ 0\ \ 0\ \ 0)^T$, changing the second column to this vector, and finally solving with the right-hand side $(8\ \ 45\ \ 31\ \ 15\ \ 17)^T$ and the matrix transposed.

Suitable code is as follows:–

```
C SIMPLE EXAMPLE OF USE OF LA15 PACKAGE
      INTEGER    IND(30,2),IP(20),IW(80)
      LOGICAL TRANS
      DOUBLE PRECISION A(30),W(10),G,U,B(10)
      DOUBLE PRECISION CNTL(3)
      INTEGER ICNTL(3)
      INTEGER KEEP(7)
C
C INITIALISE LA15
      CALL LA15ID(ICNTL,CNTL,KEEP)
C
C STORE VALUES FOR IA, U AND TRANS
      IA = 30
      U =0.1D0
      TRANS = .FALSE.
C
C READ MATRIX
      READ (5,*) N,NZ
      READ (5,*) (IND(I,1),IND(I,2),A(I),I=1,NZ)
C
C FACTORIZE MATRIX
      CALL LA15AD(A,IND,NZ,IA,N,IP,IW,W,G,U,ICNTL,CNTL,KEEP)
C
C READ NEW COLUMN
      READ (5,*) M,(B(I),I=1,N)
C
C SOLVE EQUATION
      CALL LA15BD(A,IND,IA,N,IP,IW,W,G,B,TRANS,ICNTL,KEEP)
C
C PRINT SOLUTION
      WRITE(6,10) G,(B(I),I=1,N)
 10   FORMAT(/' G=',E10.2/' SOLUTION=', 5F10.5)
C
```

```
C UPDATE FACTORIZATION
      CALL LA15CD(A,IND,IA,N,IP,IW,W,G,U,M,ICNTL,CNTL,KEEP)
C
C READ RIGHT-HAND SIDE AND SET TRANS
      READ (5,*) (B(I),I=1,N)
      TRANS = .TRUE.
C
C SOLVE EQUATION
      CALL LA15BD(A,IND,IA,N,IP,IW,W,G,B,TRANS,ICNTL,KEEP)
C
C PRINT SOLUTION
      WRITE(6,10) G,(B(I),I=1,N)
C
      STOP
      END
```

Suitable data is as follows:–

```
 5 8
 1 1 2.0
 2 2 3.0
 2 3 4.0
 2 5 6.0
 3 3 1.0
 3 4 5.0
 4 3 5.0
 5 5 1.0
 2 1.0 4.0 0.0 0.0 0.0
 8. 45. 31. 15. 17.
```

This produces the following output

```
 G=  0.60E+01
 SOLUTION=   0.50000   1.33333   0.00000   0.00000   0.00000

 G=  0.60E+01
 SOLUTION=   4.00000  10.25000   3.00000  -2.60000 -44.50000
```