1 SUMMARY

To write a sparse matrix in Harwell-Boeing format [1] with possible right-hand sides. The matrix can be input as an assembled matrix in either column-oriented or coordinate form, or as an unassembled finite-element matrix. The right-hand sides must be in full format.


2 HOW TO USE THE PACKAGE

2.1 The argument list

There is a single subroutine for both assembled and unassembled matrices, with or without right-hand sides.

The single precision version

```fortran
CALL MC52A(ICNTL,TITLE,KEY,M,N,NE,IP,IND,ASPK,VALFMT,
* RHS,RHSFMT,IW)
```

The double precision version

```fortran
CALL MC52AD(ICNTL,TITLE,KEY,M,N,NE,IP,IND,ASPK,VALFMT,
* RHS,RHSFMT,IW)
```

ICNTL is an INTEGER array of length 10 whose first 5 components must be set by the user. It holds information on the input matrix and output file and controls the action of the subroutine. This argument is not altered by the subroutine.

ICNTL(1) must be set by the user to indicate whether only the pattern is being supplied (ICNTL(1)=0) or whether real values are supplied (ICNTL(1)≠0).

ICNTL(2) must be set by the user to indicate whether the matrix is symmetric (ICNTL(2)=0), skew symmetric (ICNTL(2)=1), rectangular (ICNTL(2)=2), or unsymmetric (ICNTL(2)>2).

ICNTL(3) must be set by the user to indicate the format of the input matrix: ICNTL(3)=0 for an assembled matrix in column-oriented form, ICNTL(3)=1 for an assembled matrix in coordinate form, and ICNTL(3)≠0 or 1 for an unassembled finite-element matrix.

ICNTL(4) must be set by the user to the number of right-hand sides.

ICNTL(5) must be set by the user to the stream number for the output file on which the Harwell-Boeing format will be written.

ICNTL(6) to ICNTL(10) are not accessed by the routine.

TITLE is a CHARACTER*72 variable that must be set by the user to the text for the title line for the matrix. This argument is not altered by the subroutine.

KEY is a CHARACTER*8 variable that must be set by the user to the identifying key for the matrix. This should be distinct from that of any other matrix. Current keys can be found in the Users’ Guide to the Harwell-Boeing Collection. This argument is not altered by the subroutine.
is an INTEGER variable that must be set by the user. If the matrix is assembled, \( M \) is the number of rows of the matrix; if the matrix is an unassembled finite-element matrix, \( M \) is the value of the largest index used. This argument is not altered by the subroutine.

\( N \) is an INTEGER variable that must be set by the user. If the matrix is assembled, \( N \) is the number of columns of the matrix; if the matrix is an unassembled finite-element matrix, \( N \) is the number of elements. This argument is not altered by the subroutine.

\( NE \) is an INTEGER variable that must be set by the user. If the matrix is assembled, \( NE \) is the number of entries in the matrix; if the matrix is an unassembled finite-element matrix, \( NE \) is the number of indices in the element lists. This argument is not altered by the subroutine.

\( IP \) is an INTEGER array of size \( N+1 \). If the matrix is stored in column-oriented form, \( IP \) must be set by the user so that \( IP(J), J=1, \ldots, N \) holds the position in \( IND \) of the first entry in column \( J \) and \( IP(N+1) \) holds the position of the last entry in column \( N \). For unassembled finite-element problems (\( ICNTL(3) \neq 0 \) or \( 1 \)), \( IP \) must point to the start of each element in array \( IND \). In these cases, this argument is not altered by the subroutine. If the matrix is held in coordinate form, \( IP \) need not be set on entry and, on exit, it will be set to column pointers for the reordered matrix.

\( IND \) is an INTEGER array whose first \( NE \) positions must be set by the user. If the matrix is assembled, \( IND \) must hold the row indices of the entries. If storage is column-oriented, these must be ordered by columns. If storage is in coordinate form, the corresponding column indices must be in positions \( IND(NE+I), I=1, \ldots, NE \). In this case, the matrix entries can be in any order. For a symmetric matrix, only the lower triangle should be stored. If the matrix is an unassembled finite-element matrix, \( IND \) must hold the list of indices of the variables in each element in turn. \( IND \) is not altered by the subroutine, unless input is in coordinate form (\( ICNTL(3)=1 \)), in which case it will hold the row indices of the reordered matrix.

\( ASPK \) is a REAL (DOUBLE PRECISION in the D version) array. If \( ICNTL(1) \neq 0 \), it must be set by the user to contain the numerical values of the entries. For an assembled matrix, the first \( NE \) positions must contain the values of the entries whose row indices are in corresponding positions in \( IND \). For an unassembled finite-element matrix, the first entries of \( ASPK \) must contain contiguously the numerical values of each element. If these are symmetric, only the lower triangle should be held ordered by columns. \( ASPK \) is not altered by the subroutine, unless the matrix is input in coordinate form (\( ICNTL(3)=1 \)), in which case it will hold the values for the reordered matrix.

\( VALFMT \) is a CHARACTER*20 variable that must be set by the user only if reals are input (\( ICNTL(1) \neq 0 \)). It should be set to the format for the reals. Normally floating-point format is used, for example \( '(4D20.8)' \).

\( RHS \) is a REAL (DOUBLE PRECISION in the D version) array whose first \( M*ICNTL(4) \) entries must be set by the user to the right-hand side values. For multiple right-hand sides, each is stored contiguously. \( RHS \) is not accessed if \( ICNTL(4)=0 \). This argument is not altered by the subroutine.

\( RHSFMT \) is a CHARACTER*20 variable that must be set by the user only if right-hand sides are input (\( ICNTL(4) > 0 \)). It should be set to the format for the right-hand sides. Normally floating-point format is used, and the user may want a lower precision than for the reals, for example \( '(5D16.5)' \).

\( IW \) is an INTEGER work array of length \( N+1 \) that is only accessed if the matrix is input in coordinate form (\( ICNTL(3)=1 \)).

### 2.2 Error diagnostics

There are no error diagnostics.

### 3 GENERAL INFORMATION

#### Workspace:

\( IW \) of length \( N+1 \). Only used if \( ICNTL(3)=1 \).

#### Use of common:

None.
Other routines called directly: MC52A/AD calls internal routines MA51B/BD and MA51C/CD that need not be called by the user. The sorting routine MC49A/AD is also called (only if ICNTL(3) = 1).

Input/output: None.

4 METHOD

The code calculates data values for the header lines and then outputs them and the input data to the file ICNTL(5), first sorting the matrix to column order if input is in coordinate form.

5 EXAMPLES OF USE

We illustrate the use of the subroutine by a small program that reads a matrix from a file and outputs it in Harwell-Boeing format.

C This program reads a matrix from the file on stream 5, then outputs
C the Harwell-Boeing format to stream 9. This file is then printed on
C unit LP.

PROGRAM SPEC

INTEGER NMAX,NZMAX,NZRHS
PARAMETER (NMAX=100, NZMAX=4000, NZRHS=1)

INTEGER ICNTL(10)
CHARACTER TITLE*72, KEY*8
INTEGER M, N, NE, IP(NMAX+1), IND(NZMAX)
DOUBLE PRECISION ASPK(NZMAX)
CHARACTER*20 VALFMT
DOUBLE PRECISION RHS(NZRHS)
CHARACTER*20 RHSFMT
INTEGER IW(NMAX+1)

C Local variables
CHARACTER*80 CARD
INTEGER I, LP

C Set output stream for printing
LP = 6

C Set ICNTL
ICNTL(1) = 0
ICNTL(2) = 3
ICNTL(3) = 0
ICNTL(4) = 0
ICNTL(5) = 9

C Read title, key, and matrix order
READ(5,'(A72)') TITLE
READ(5,'(A8)') KEY
READ(5,*) M,N

C Read pointer array
READ(5,*) (IP(I),I=1,N+1)
NE = IP(N+1) - 1

C Read reals
READ(5,*) (IND(I),I=1,NE)
C Call MC52
   CALL MC52AD(ICNTL,TITLE,KEY,M,N,NE,IP,IND,
       ASPK,VALFMT,RHS,RHSFMT,IW)

C The output file is now printed
   REWIND ICNTL(5)
   20 READ(ICNTL(5),*(A80),END=100) CARD
   WRITE(LP,'*(A80)') CARD
   GO TO 20

   100 STOP
   END

With the data:

1U JOHN G. LEWIS P4 COUNTEREXAMPLE WHICH REQUIRES FILL IN SPIKES
JGL009
   9 9
   1 9 13 21 27 33 39 44 46 51
   1 2 4 5 6 7 8 9 2 3 4 5 6 7 8 9 4 5 6 7 8 9 4 5 6 7 8 9 4 5 6 7 8
   0 9 1 2 3 8 9 1 2 3 8 9

this program produces:

1U JOHN G. LEWIS P4 COUNTEREXAMPLE WHICH REQUIRES FILL IN SPIKES
JGL009
   4 1 3 0 0
   9 9
   0 (20I4) (20I4)
   1 9 13 21 27 33 39 44 46 51
   1 2 4 5 6 7 8 9 2 3 4 5 6 7 8 9 2 3 4 5 6 7 8 9
   4 5 6 7 8 9 4 5 6 7 8 9 4 5 6 7 8 9 1 2
   3 8 9 8 9 1 2 3 8 9