1 SUMMARY

Given the pattern of nonzeros of a sparse matrix, this subroutine attempts to find a row permutation that makes the matrix have nonzeros on its diagonal.


2 HOW TO USE THE PACKAGE

2.1 Argument list

The single precision version

CALL MC21A(N, ICN, LICN, IP, LENR, IPERM, NUMNZ, IW)

The double precision version

CALL MC21AD(N, ICN, LICN, IP, LENR, IPERM, NUMNZ, IW)

N is an INTEGER variable which must be set by the user to the order of the matrix. It is not altered by MC21A/AD.

ICN is an INTEGER array of length LICN which must be set by the user to contain the column indices of the nonzeros. Those belonging to a single row must be contiguous but the ordering of column indices within each row is unimportant and wasted space between rows is permitted. It is not altered by MC21A/AD.

LICN is an INTEGER which must be set by the user to the length of array ICN. It is not altered by MC21A/AD.

IP is an INTEGER array of length N and must be set by the user so that IP(I) contains the position in array ICN of the first column index of a nonzero in row I, for I=1, 2,..., N. It is not altered by MC21A/AD.

LENR is an INTEGER array of length N. The user must set LENR(I) equal to the number of nonzeros in row I, I=1,2,..., N. It is not altered by MC21A/AD.

IPERM is an INTEGER array of length N in which the row permutation is output. IPERM(I) gives the position in the original matrix of row I in the permuted matrix, I=1,...,N.

NUMNZ is an INTEGER output variable which gives the number of nonzeros on the diagonal of the reordered matrix. If this is less than N, the matrix is structurally singular and will be a fortiori numerically singular. For an example of this, see Section 2.2.

IW is an INTEGER work array length at least 4*N.

2.2 Error returns

There are no error returns. However, the user may input a matrix for which there is no permutation that makes the diagonal zero-free. An example of this is

\[
\begin{bmatrix}
\times & 0 \\
\times & 0 \\
\end{bmatrix}
\]

In such instances the algorithm will produce a permutation which will put as many nonzeros on the diagonal as possible (1 in the above example). This number will be output in NUMNZ. The array IPERM will still hold a permutation of the integers 1, 2,..., N but in this case N-NUMNZ of the elements (IPERM(I), I) will be zero.
3 GENERAL INFORMATION

Workspace: See argument IW.
Use of common: None.
Other routines called directly: MC21A/AD calls MC21B/BD which never needs to be called directly by the user.
Input/output: None.
Restrictions: None.

4 METHOD


5 EXAMPLE OF USE

An example to permute a matrix so as to remove zeros from the diagonal

In the example code shown below, we read in the entries of a sparse matrix by rows. We then call the routine MC21A to find a row permutation that will minimise the number of zeros on the diagonal.

```fortran
INTEGER MAXN, MAXNZ, LICN
PARAMETER( MAXN = 5, MAXNZ = MAXN*MAXN, LICN = 4*MAXNZ )
INTEGER ICN(LICN), N, I, J, NUMNZ,
+ LENR(MAXN), IP(MAXN), IPERM(MAXN), IW(4*MAXNZ)
READ(5, *) N
C IP(I) POINTS TO THE START OF THE I'TH ROW OF THE MATRIX
C LENR(I) HOLDS THE NUMBER OF ENTRIES IN THE I'TH ROW
IP(1) = 1
DO 10 I=1,N
   READ(5, *) LENR(I), (ICN(J),J=IP(I),IP(I)+LENR(I)-1)
   IP(I+1) = IP(I) + LENR(I)
10 CONTINUE
CALL MC21A( N, ICN, LICN, IP, LENR, IPERM, NUMNZ, IW)
IF (NUMNZ.LT.N) WRITE(6,20) N-NUMNZ
WRITE(6,30) (IPERM(I),I=1,N)
20 FORMAT(/© WARNING, THE MATRIX IS STRUCTURALLY SINGULAR©,/, + © THE NUMBER OF ZERO ENTRIES ON THE DIAGONAL IS ©,I2 /)
30 FORMAT( / © THE PERMUTATION ARRAY IPERM() IS:© , 4I2)
STOP
END
```

Thus for the matrix

\[
\begin{pmatrix}
1 & 0 & 0 & 4 \\
0 & 0 & 7 & 8 \\
9 & 0 & 0 & 12 \\
0 & 14 & 0 & 16 \\
\end{pmatrix}
\]

we could have as input

4
2 1 4
2 4 3
2 4 1
2 2 4

and we would get the following output
The permutation array IPERM() is: 1 4 2 3

The array IPERM() represents the permutation matrix
\[ P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \]

and using this permutation our input matrix
\[ \begin{pmatrix} 1 & 0 & 0 & 4 \\ 0 & 14 & 0 & 16 \\ 0 & 0 & 7 & 8 \\ 9 & 0 & 0 & 12 \end{pmatrix} \]

would be transformed to...