

NAG Library Routine Document

F03ADF

Note: before using this routine, please read the Users' Note for your implementation to check the interpretation of *bold italicised* terms and other implementation-dependent details.

1 Purpose

F03ADF calculates the determinant of a complex matrix using an *LU* factorization with partial pivoting.

2 Specification

```
SUBROUTINE F03ADF (A, LDA, N, DETR, DETI, WKSPACE, IFAIL)
```

```
INTEGER          LDA, N, IFAIL
REAL (KIND=nag_wp)  DETR, DETI, WKSPACE(N)
COMPLEX (KIND=nag_wp) A(LDA,*)
```

3 Description

F03ADF calculates the determinant of A using the *LU* factorization with partial pivoting, $PA = LU$, where P is a permutation matrix, L is lower triangular and U is unit upper triangular. The determinant of A is the product of the diagonal elements of L , with the correct sign determined by the row interchanges.

4 References

Wilkinson J H and Reinsch C (1971) *Handbook for Automatic Computation II, Linear Algebra* Springer-Verlag

5 Parameters

1: A(LDA,*) – COMPLEX (KIND=nag_wp) array *Input/Output*

Note: the second dimension of the array A must be at least $\max(1, N)$.

On entry: the n by n matrix A .

On exit: A is overwritten by the factors L and U , except that the unit diagonal elements of U are not stored.

2: LDA – INTEGER *Input*

On entry: the first dimension of the array A as declared in the (sub)program from which F03ADF is called.

Constraint: $LDA \geq \max(1, N)$.

3: N – INTEGER *Input*

On entry: n , the order of the matrix A .

Constraint: $N \geq 0$.

4: DETR – REAL (KIND=nag_wp) *Output*

5: DETI – REAL (KIND=nag_wp) *Output*

On exit: the real part and imaginary parts of the determinant of A .

6: WKSPACE(N) – REAL (KIND=nag_wp) array *Workspace*

7: IFAIL – INTEGER *Input/Output*

On entry: IFAIL must be set to 0, –1 or 1. If you are unfamiliar with this parameter you should refer to Section 3.3 in the Essential Introduction for details.

For environments where it might be inappropriate to halt program execution when an error is detected, the value –1 or 1 is recommended. If the output of error messages is undesirable, then the value 1 is recommended. Otherwise, if you are not familiar with this parameter, the recommended value is 0. **When the value –1 or 1 is used it is essential to test the value of IFAIL on exit.**

On exit: IFAIL = 0 unless the routine detects an error or a warning has been flagged (see Section 6).

6 Error Indicators and Warnings

If on entry IFAIL = 0 or –1, explanatory error messages are output on the current error message unit (as defined by X04AAF).

Errors or warnings detected by the routine:

IFAIL = 1

The matrix A is singular, possibly due to rounding errors. The factorization could not be completed. DETR and DETI are set to 0.0.

IFAIL = 2

Overflow. The value of the determinant is too large to be held in the computer.

IFAIL = 3

Underflow. The value of the determinant is too small to be held in the computer.

IFAIL = 4

On entry, $N < 0$,
or $LDA < \max(1, N)$.

7 Accuracy

The accuracy of the determinant depends on the conditioning of the original matrix. For a detailed error analysis, see page 107 of Wilkinson and Reinsch (1971).

8 Further Comments

The time taken by F03ADF is approximately proportional to n^3 .

9 Example

This example calculates the determinant of the complex matrix

$$\begin{pmatrix} 1 & 1 + 2i & 2 + 10i \\ 1 + i & 3i & -5 + 14i \\ 1 + i & 5i & -8 + 20i \end{pmatrix}.$$

9.1 Program Text

```

Program f03adfe

!      F03ADF Example Program Text
!
!      Mark 24 Release. NAG Copyright 2012.
!
!      .. Use Statements ..
!      Use nag_library, Only: f03adf, nag_wp
!      .. Implicit None Statement ..
!      Implicit None
!      .. Parameters ..
!      Integer, Parameter          :: nin = 5, nout = 6
!      .. Local Scalars ..
!      Real (Kind=nag_wp)          :: deti, detr
!      Integer                     :: i, ifail, lda, n
!      .. Local Arrays ..
!      Complex (Kind=nag_wp), Allocatable :: a(:, :)
!      Real (Kind=nag_wp), Allocatable   :: wkspce(:)
!      .. Executable Statements ..
!      Write (nout,*) 'F03ADF Example Program Results'

!      Skip heading in data file
!      Read (nin,*)

!      Read (nin,*) n
!      lda = n
!      Allocate (a(lda,n),wkspce(n))

!      Read (nin,*)(a(i,1:n),i=1,n)

!      ifail = 0
!      Call f03adf(a,lda,n,detr,deti,wkspce,ifail)

!      Write (nout,*)
!      Write (nout,99999) detr, deti

99999 Format (1X,'Value of determinant = (',F9.4,',',F9.4,')')
End Program f03adfe

```

9.2 Program Data

```

F03ADF Example Program Data
3
(1.0, 0.0) (1.0, 2.0) (2.0,10.0)
(1.0, 1.0) (0.0, 3.0) (-5.0,14.0)
(1.0, 1.0) (0.0, 5.0) (-8.0,20.0)

```

9.3 Program Results

```

F03ADF Example Program Results

Value of determinant = ( 1.0000, 0.0000)

```
