

# NAG Library Routine Document

## D02NSF

**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

D02NSF is a setup routine which must be called prior to an integrator in Sub-chapter D02M–N, if full matrix linear algebra is required.

### 2 Specification

```
SUBROUTINE D02NSF (NEQ, NEQMAX, JCEVAL, NWKJAC, RWORK, IFAIL)
  INTEGER          NEQ, NEQMAX, NWKJAC, IFAIL
  REAL (KIND=nag_wp) RWORK(50+4*NEQMAX)
  CHARACTER(1)    JCEVAL
```

### 3 Description

D02NSF defines the linear algebra to be used as full matrix linear algebra, permits you to specify the method for calculating the Jacobian and checks the validity of certain input values.

### 4 References

See the D02M–N Sub-chapter Introduction.

### 5 Arguments

- 1: NEQ – INTEGER *Input*  
*On entry:* the number of differential equations.  
*Constraint:*  $1 \leq \text{NEQ} \leq \text{NEQMAX}$ .
- 2: NEQMAX – INTEGER *Input*  
*On entry:* a bound on the maximum number of differential equations to be solved during the integration.  
*Constraint:*  $\text{NEQMAX} \geq \text{NEQ}$ .
- 3: JCEVAL – CHARACTER(1) *Input*  
*On entry:* specifies the technique to be used to compute the Jacobian.  
 JCEVAL = 'N'  
 The Jacobian is to be evaluated numerically by the integrator. If this option is used, then the actual argument corresponding to JAC in the call to D02NBF or D02NGF must be either D02NBZ or D02NGZ respectively.  
 JCEVAL = 'A'  
 You must supply a (sub)program to evaluate the Jacobian on a call to the integrator.  
 JCEVAL = 'D'  
 The default choice is to be made. In this case 'D' is interpreted as 'N'.

Only the first character of the actual argument JCEVAL is passed to D02NSF; hence it is permissible for the actual argument to be more descriptive 'Numerical', 'Analytical' or 'Default' on a call to D02NSF.

*Constraint:* JCEVAL = 'N', 'A' or 'D'.

4: NWKJAC – INTEGER

*Input*

*On entry:* the size of the workspace array WKJAC, which you are supplying to the integrator, as declared in the (sub)program from which D02NSF is called.

*Constraint:*  $NWKJAC \geq NEQMAX \times (NEQMAX + 1)$ .

5: RWORK(50 + 4 × NEQMAX) – REAL (KIND=nag\_wp) array

*Communication Array*

This must be the same workspace array as the array RWORK supplied to the integrator. It is used to pass information from the setup routine to the integrator and therefore the contents of this array must not be changed before calling the integrator.

6: IFAIL – INTEGER

*Input/Output*

*On entry:* IFAIL must be set to 0, -1 or 1. If you are unfamiliar with this argument you should refer to Section 3.4 in How to Use the NAG Library and its Documentation 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 argument, 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

On entry, NEQ < 1,  
or NEQ > NEQMAX,  
or  $NWKJAC < NEQMAX \times (NEQMAX + 1)$ ,  
or JCEVAL ≠ 'N', 'A' or 'D'.

IFAIL = -99

An unexpected error has been triggered by this routine. Please contact NAG.

See Section 3.9 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -399

Your licence key may have expired or may not have been installed correctly.

See Section 3.8 in How to Use the NAG Library and its Documentation for further information.

IFAIL = -999

Dynamic memory allocation failed.

See Section 3.7 in How to Use the NAG Library and its Documentation for further information.

## **7 Accuracy**

Not applicable.

## **8 Parallelism and Performance**

D02NSF is not thread safe and should not be called from a multithreaded user program. Please see Section 3.12.1 in *How to Use the NAG Library and its Documentation* for more information on thread safety.

D02NSF is not threaded in any implementation.

## **9 Further Comments**

D02NSF must be called as a setup routine before a call to either D02NBF or D02NGF and may be called as the linear algebra setup routine before a call to either D02NMF or D02NNF.

## **10 Example**

See Section 10 in D02NBF, D02NGF and D02NMF.

---