

NAG Fortran Library Routine Document

E04NHF/E04NHA

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

To supply individual optional parameters to E04NFF/E04NFA. More precisely, E04NHF must be used to supply optional parameters to E04NFF and E04NHA must be used to supply optional parameters to E04NFA.

E04NHA is a version of E04NHF that has additional parameters in order to make it safe for use in multithreaded applications (see Section 5 below). The initialisation routine E04WBF **must** have been called prior to calling E04NHA.

2 Specifications

2.1 Specification for E04NHF

```
SUBROUTINE E04NHF (STRING)
CHARACTER*(*)    STRING
```

2.2 Specification for E04NHA

```
SUBROUTINE E04NHA (STRING, LWSAV, IWSAV, RWSAV, INFORM)
INTEGER          IWSAV(610), INFORM
real           RWSAV(475)
LOGICAL         LWSAV(120)
CHARACTER*(*)   STRING
```

3 Description

E04NHF/E04NHA may be used to supply values for optional parameters to the corresponding routines E04NFF/E04NFA. It is only necessary to call E04NHF/E04NHA for those parameters whose values are to be different from their default values. One call to E04NHF/E04NHA sets one parameter value.

Each optional parameter is defined by a single character string, of up to 72 characters, consisting of one or more items. The items associated with a given option must be separated by spaces, or equals signs [=]. Alphabetic characters may be upper or lower case. The string

```
Print level = 1
```

is an example of a string used to set an optional parameter. For each option the string contains one or more of the following items:

- (a) A mandatory keyword.
- (b) A phrase that qualifies the keyword.
- (c) A number that specifies an INTEGER or *real* value. Such numbers may be up to 16 contiguous characters in Fortran's I, F, E or D formats, terminated by a space if this is not the last item on the line.

Blank strings and comments are ignored. A comment begins with an asterisk (*) and all subsequent characters in the string are regarded as part of the comment.

For E04NHF, each user-specified option is normally printed as it is defined, on the current advisory message unit (see X04ABF), but this printing may be suppressed using the keyword **nolist**. Thus the statement

```
CALL E04NHF ('Nolist')
```

suppresses printing of this and subsequent options. Printing will automatically be turned on again after a call to E04NFF and may be turned on again at any time using the keyword **list**.

For E04NHA printing is turned off by default, but may be turned on at any time using the keyword **list**.

Optional parameter settings are preserved following a call to E04NFF/E04NFA and so the keyword **defaults** is provided to allow you to reset all the optional parameters to their default values prior to a subsequent call to E04NFF/E04NFA.

A complete list of optional parameters, their abbreviations, synonyms and default values is given in Section 11 of the document for E04NFF/E04NFA.

4 References

None.

5 Parameters

1: STRING – CHARACTER*(*) *Input*

On entry: a single valid option string (as described in Section 3 above and in Section 11 of the document for E04NFF/E04NFA).

Note: *the following are additional parameters for specific use with E04NHA. Users of E04NHF therefore need not read the remainder of this section.*

2: LWSAV(120) – LOGICAL array *Workspace*

3: IWSAV(610) – INTEGER array *Workspace*

4: RWSAV(475) – *real* array *Workspace*

The arrays LWSAV, IWSAV and RWSAV **must not** be altered between calls to any of the routines E04WBF, E04NKA, E04NLA or E04NHA.

5: INFORM – INTEGER *Output*

On exit: contains zero if a valid option string has been supplied and a value > 0 otherwise (see Section 6).

6 Error Indicators and Warnings

Errors or warnings detected by the routine:

INFORM = 5

The supplied option is invalid. Check that the keywords are neither ambiguous nor misspelt.

7 Accuracy

Not applicable.

8 Further Comments

E04NGF/E04NGA may also be used to supply optional parameters to the corresponding routines E04NFF/E04NFA.

9 Example

See Section 9 of the document for E04NGF/E04NGA.