NAG Library Routine Document

E04ZNF

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

E04ZNF is an option getting routine for solvers from the NAG optimization modelling suite, namely E04STF and E04SVF. It is used to query the value of optional parameters.

2 Specification

```
SUBROUTINE E04ZNF (HANDLE, OPTSTR, IVALUE, RVALUE, CVALUE, OPTYPE,         &
                   IFAIL)

INTEGER            IVALUE, OPTYPE, IFAIL
REAL (KIND=nag_wp) RVALUE
CHARACTER(*)       OPTSTR, CVALUE
TYPE (C_PTR)       HANDLE
```

3 Description

E04ZNF is used to query the current values of options. It can be especially useful to retrieve the optional parameters left for automatic choice by the solver.

This routine will normally return either an integer, real or character value dependent upon the type associated with the optional parameter being queried. This is indicated by the returned value of OPTYPE.

Information relating to available option names is given in the documentation of the particular solver. See also E04RAF for a generic description of the suite.

4 References

None.

5 Arguments

1: HANDLE – TYPE (C_PTR) *Input*

On entry: the handle to the problem. It needs to be initialized by E04RAF and **must not** be changed.

2: OPTSTR – CHARACTER(*) *Input*

On entry: a string identifying the option whose current value is required. See the documentation of the particular solver for information on valid options.

3: IVALUE – INTEGER *Output*

On exit: if the optional parameter supplied in OPTSTR is an integer-valued argument, IVALUE will hold its current value.

4: RVALUE – REAL (KIND=nag_wp) *Output*

On exit: if the optional parameter supplied in OPTSTR is a real-valued argument, RVALUE will hold its current value.

5: CVALUE – CHARACTER(*) *Output*

Note: the string returned in CVALUE will never exceed 40 characters in length.

On exit: if the optional parameter supplied in OPTSTR is a character-valued argument, CVALUE will hold its current value.

6: OPTYPE – INTEGER *Output*

On exit: indicates whether the optional parameter supplied in OPTSTR is an integer, real or character-valued argument and hence which of IVALUE, RVALUE or CVALUE holds the current value.

OPTYPE = 1
OPTSTR is an integer-valued optional parameter; its current value has been returned in IVALUE.

OPTYPE = 2
OPTSTR is a real-valued optional parameter; its current value has been returned in RVALUE.

OPTYPE = 3
OPTSTR is a character-valued optional parameter; its current value has been returned in CVALUE.

7: 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

The supplied HANDLE does not define a valid handle to the data structure for the NAG optimization modelling suite. It has not been initialized by E04RAF or it has been corrupted.

IFAIL = 11

On entry, the *option* supplied in OPTSTR was not recognized: OPTSTR = ⟨*value*⟩.

IFAIL = 41

On entry, OPTSTR indicates a character optional parameter, but CVALUE is too short to hold the stored value. The returned value will be truncated.

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

E04ZNF is not threaded in any implementation.

9 Further Comments

None.

10 Example

See the example program associated with the solver E04SVF for a demonstration of how to use E04ZNF to query options. See also Section 10 in E04RAF for links to all examples in the suite.