NAG Library Routine Document

D01ZKF

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

D01ZKF either initializes or resets the optional parameter arrays or sets a single optional parameter for supported problem solving routines in Chapter D01.

2 Specification

```
SUBROUTINE D01ZKF (OPTSTR, IOPTS, LIOPTS, OPTS, LOPTS, IFAIL)
INTEGER            IOPTS(LIOPTS), LIOPTS, LOPTS, IFAIL
REAL (KIND=nag_wp) OPTS(LOPTS)
CHARACTER(*)       OPTSTR
```

3 Description

D01ZKF has three purposes: to initialize optional parameter arrays; to reset all optional parameters to their default values; or to set a single optional parameter to a user-supplied value.

Optional parameters and their values are, in general, presented as a character string, OPTSTR, of the form '*option* = *optval*'; alphabetic characters can be supplied in either upper or lower case. Both *option* and *optval* may consist of one or more tokens separated by white space. The tokens that comprise *optval* will normally be either an integer, real or character value as defined in the description of the specific optional argument. In addition all optional parameters can take an *optval* DEFAULT which resets the optional parameter to its default value.

It is imperative that optional parameter arrays are initialized before any options are set, before the relevant problem solving routine is called and before any options are queried using D01ZLF. To initialize the optional parameter arrays IOPTS and OPTS for a specific problem solving routine, the option **Initialize** is used with *optval* identifying the problem solving routine to be called, via its short name. For example, to initialize the optional parameter arrays to be passed to D01RAF and its associated routine D01RCF, D01ZKF is called as follows:

```
call D01ZKF('Initialize = d01raf', IOPTS, LIOPTS, OPTS, LOPTS, IFAIL)
```

The available option names and their corresponding valid values are given in Section 11 in D01ESF and D01RAF.

4 References

None.

5 Parameters

1: OPTSTR – CHARACTER(*) *Input*

On entry: a string identifying the option to be set.

Initialize = *routine name*
Initialize the optional parameter arrays IOPTS and OPTS for use with routine *routine name*, where *routine name* is the short name associated with the routine of interest.

Defaults
Resets all options to their default values.

option = *optval*
See Section 11 in D01ESF and D01RAF for details of valid values for *option* and *optval*. The equals sign (=) delimiter must be used to separate the *option* from its *optval* value.

OPTSTR is case insensitive. Each token in the *option* and *optval* component must be separated by at least one space.

2: IOPTS(LIOPTS) – INTEGER array *Communication Array*

On entry: optional parameter array.

If OPTSTR has the form **Initialize** = *routine name*, the contents of IOPTS need not be set.

Otherwise, IOPTS **must not** have been altered since the last call to D01ZKF, D01ZLF or the selected problem solving routine.

On exit: dependent on the contents of OPTSTR, either an initialized, reset or updated version of the optional parameter array.

3: LIOPTS – INTEGER *Input*

On entry: the length of the array IOPTS.

Constraint: unless otherwise stated in the documentation for a specific, supported, problem solving routine, LIOPTS ≥ 100.

4: OPTS(LOPTS) – REAL (KIND=nag_wp) array *Communication Array*

On entry: optional parameter array.

If OPTSTR has the form **Initialize** = *routine name*, the contents of OPTS need not be set.

Otherwise, OPTS **must not** have been altered since the last call to D01ZKF, D01ZLF or the selected problem solving routine.

On exit: dependent on the contents of OPTSTR, either an initialized, reset or updated version of the optional parameter array.

5: LOPTS – INTEGER *Input*

On entry: the length of the array OPTS.

Constraint: unless otherwise stated in the documentation for a specific, supported, problem solving routine, LOPTS ≥ 100.

6: 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 $= 11$

On entry, the optional parameter in OPTSTR was not recognized: OPTSTR $= \langle value \rangle$.

IFAIL $= 12$

On entry, the expected delimiter '=' was not found in OPTSTR: OPTSTR $= \langle value \rangle$.

IFAIL $= 13$

On entry, could not convert the specified *optval* to an integer: OPTSTR $= \langle value \rangle$.

On entry, could not convert the specified *optval* to a real: OPTSTR $= \langle value \rangle$.

IFAIL $= 14$

On entry, attempting to initialize the optional parameter arrays but specified routine name was not valid: name $= \langle value \rangle$.

IFAIL $= 15$

On entry, the *optval* supplied for the integer optional parameter is not valid.
OPTSTR $= \langle value \rangle$.

IFAIL $= 16$

On entry, the *optval* supplied for the real optional parameter is not valid.
OPTSTR $= \langle value \rangle$.

IFAIL $= 17$

On entry, the *optval* supplied for the character optional parameter is not valid.
OPTSTR $= \langle value \rangle$.

IFAIL $= 21$

On entry, either the option arrays have not been initialized or they have been corrupted.

IFAIL $= 31$

On entry, LIOPTS $= \langle value \rangle$.
Constraint: LIOPTS $\geq \langle value \rangle$.

IFAIL $= 51$

On entry, LOPTS $= \langle value \rangle$.
Constraint: LOPTS $\geq \langle value \rangle$.

IFAIL $= -99$

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

See Section 3.8 in the Essential Introduction for further information.

IFAIL $= -399$

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

See Section 3.7 in the Essential Introduction for further information.

IFAIL $= -999$

Dynamic memory allocation failed.

See Section 3.6 in the Essential Introduction for further information.

7 Accuracy

Not applicable.

8 Parallelism and Performance

D01ZKF is threaded by NAG for parallel execution in multithreaded implementations of the NAG Library.

Please consult the X06 Chapter Introduction for information on how to control and interrogate the OpenMP environment used within this routine. Please also consult the Users' Note for your implementation for any additional implementation-specific information.

9 Further Comments

For suites of routines that share the same option arrays, the option arrays must be initialized using the primary (driver) routine name. For example for routines D01RAF and D01RCF, the option arrays must be initialized for D01RAF.

When encoding integer valued options in OPTSTR, the integer *optval* must be written as an explicit integer. For example, `"Maximum Subdivisions = 12"` is acceptable, whereas `"Maximum Subdivisions = 12.0"` and `"Maximum Subdivisions = 0.12E2"` are not.

When encoding real valued options in OPTSTR, the *optval* may be integral if appropriate. For example, `"Absolute Tolerance = 10"`, `"Absolute Tolerance = 10.0"` and `"Absolute Tolerance = 1.0E1"` are all acceptable.

10 Example

See the example programs associated with the problem solving routine you wish to use for a demonstration of how to use D01ZKF to initialize option arrays and set options.