

NAG Library Routine Document

G02ZKF

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

G02ZKF either initializes or resets the optional parameter arrays or sets a single optional parameter for supported problem solving routines in Chapter G02. Currently, only G02QGF is supported.

2 Specification

```
SUBROUTINE G02ZKF (OPTSTR, IOPTS, LIOPTS, OPTS, LOPTS, IFAIL)
```

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

3 Description

G02ZKF 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 G02ZLF. To initialize the optional parameter arrays IOPTS and OPTS for a specific problem solving routine, the option **Initialize** is used with *value* identifying the problem solving routine to be called, via its short name. For example, to initialize optional parameter arrays to be passed to G02QGF, G02ZKF is called as follows:

```
call G02ZKF('Initialize = g02qgf', IOPTS, LIOPTS, OPTS, LOPTS, IFAIL)
```

Information relating to available option names and their corresponding valid values is given in Section 11 in G02QGF.

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 of the problem solving routine you wish to use.

Defaults

Resets all options to their default values.

Option = *optval*

See Section 11 in G02QGF 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 G02ZKF, G02ZLF 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, $\text{LIOPTS} \geq 100$.

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 G02ZKF, G02ZLF 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, $\text{LOPTS} \geq 100$.

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 **option** supplied in OPTSTR was not recognized.

$IFAIL = 12$

On entry, the expected delimiter (=) was not found in OPTSTR.

$IFAIL = 13$

On entry, the *optval* contained in OPTSTR could not be converted into a numeric value.

$IFAIL = 14$

On entry, the option **Initialize** was identified, however the associated *optval* was not a supported routine name.

$IFAIL = 15$

On entry, the integer *optval* associated with the optional parameter is not valid.

$IFAIL = 16$

On entry, the real *optval* associated with the optional parameter is not valid.

$IFAIL = 17$

On entry, the character *optval* associated with the optional parameter is not valid.

$IFAIL = 21$

On entry, the optional parameter arrays IOPTS and OPTS have not been initialized, or have been corrupted.

$IFAIL = 31$

On entry, LIOPTS is too small.

$IFAIL = 51$

On entry, LOPTS is too small.

7 Accuracy

Not applicable.

8 Further Comments

Not applicable.

9 Example

See the example programs associated with the problem solving routine you wish to use for a demonstration of how to use G02ZKF to initialize option arrays and set options.
