

# NAG Library Function Document

## nag\_fit\_opt\_get (e02zlc)

### 1 Purpose

nag\_fit\_opt\_get (e02zlc) is used to query the value of optional arguments available to supported problem solving functions in Chapter e02. Currently, only nag\_2d\_spline\_fit\_ts\_scatter (e02jdc) is supported.

### 2 Specification

```
#include <nag.h>
#include <nage02.h>

void nag_fit_opt_get (const char *optstr, Integer *ivalue, double *rvalue,
                    char *cvalue, Integer lcvalue, Nag_VariableType *optype,
                    const Integer iopts[], const double opts[], NagError *fail)
```

### 3 Description

nag\_fit\_opt\_get (e02zlc) is used to query the current values of options. It is necessary to initialize optional argument arrays using nag\_fit\_opt\_set (e02zkc) before any options are queried.

nag\_fit\_opt\_get (e02zlc) will normally return either an integer, real or character value dependent upon the type associated with the optional argument being queried. Whether the option queried is of integer, real or character type is indicated by the returned value of **optype**.

Information on optional argument names and whether these options are real, integer or character can be found in Section 11 in nag\_2d\_spline\_fit\_ts\_scatter (e02jdc).

### 4 References

None.

### 5 Arguments

- 1: **optstr** – const char \* *Input*  
*On entry:* a string identifying the option whose current value is required. See Section 11 in nag\_2d\_spline\_fit\_ts\_scatter (e02jdc) for information on valid options. In addition, the following is a valid option:  
**Identify**  
nag\_fit\_opt\_get (e02zlc) returns in **cvalue** the function name supplied to nag\_fit\_opt\_set (e02zkc) when the optional argument arrays **iopts** and **opts** were initialized.
- 2: **ivalue** – Integer \* *Output*  
*On exit:* if the optional argument supplied in **optstr** is an integer valued argument, **ivalue** will hold its current value.
- 3: **rvalue** – double \* *Output*  
*On exit:* if the optional argument supplied in **optstr** is a real valued argument, **rvalue** will hold its current value.

- 4: **cvalue** – char \* *Output*  
**Note:** the maximum length (excluding the **NULL** terminator) of the string returned in **cvalue** depends on the problem solving routine in use. See Section 11.1 of the relevant solver.  
 The string returned in **cvalue** will never exceed **lvalue** characters in length (including the **NULL** terminator).  
*On exit:* if the optional argument supplied in **optstr** is a character valued argument, **cvalue** will hold its current value, unless **Identify** is specified (see **optstr**).
- 5: **lvalue** – Integer *Input*  
*On entry:* length of **cvalue**. At most **lvalue** – 1 non-null characters will be written into **cvalue**.  
*Constraint:* **lvalue** > 1.
- 6: **optype** – Nag\_VariableType \* *Output*  
*On exit:* indicates whether the optional argument supplied in **optstr** is an integer, real or character valued argument and hence which of **ivalue**, **rvalue** or **cvalue** holds the current value.  
**optype** = Nag\_Integer  
**optstr** is an integer valued optional argument, its current value has been returned in **ivalue**.  
**optype** = Nag\_Real  
**optstr** is a real valued optional argument, its current value has been returned in **rvalue**.  
**optype** = Nag\_Character  
**optstr** is a character valued optional argument, its current value has been returned in **cvalue**.
- 7: **iopts**[*dim*] – const Integer *Communication Array*  
**Note:** the dimension, *dim*, of this array is dictated by the requirements of associated functions that must have been previously called. This array **MUST** be the same array passed as argument **iopts** in the previous call to `nag_fit_opt_set` (e02zkc).
- 8: **opts**[*dim*] – const double *Communication Array*  
**Note:** the dimension, *dim*, of this array is dictated by the requirements of associated functions that must have been previously called. This array **MUST** be the same array passed as argument **opts** in the previous call to `nag_fit_opt_set` (e02zkc).
- 9: **fail** – NagError \* *Input/Output*  
 The NAG error argument (see Section 3.6 in the Essential Introduction).

## 6 Error Indicators and Warnings

### NE\_ALLOC\_FAIL

Dynamic memory allocation failed.  
 See Section 3.2.1.2 in the Essential Introduction for further information.

### NE\_BAD\_PARAM

On entry, argument *<value>* had an illegal value.

### NE\_INT

On entry, **lvalue** = *<value>*.  
 Constraint: **lvalue** > 1.

**NE\_INTERNAL\_ERROR**

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please contact NAG for assistance.

An unexpected error has been triggered by this function. Please contact NAG.  
See Section 3.6.6 in the Essential Introduction for further information.

**NE\_INVALID\_OPTION**

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

On entry, the optional argument in **optstr** was not recognized: **optstr** =  $\langle value \rangle$ .

**NE\_NO\_LICENCE**

Your licence key may have expired or may not have been installed correctly.  
See Section 3.6.5 in the Essential Introduction for further information.

**NW\_TRUNCATED**

On entry, **optstr** indicates a character optional argument, but **cvalue** is too short to hold the stored value. The returned value will be truncated.

**7 Accuracy**

Not applicable.

**8 Parallelism and Performance**

Not applicable.

**9 Further Comments**

None.

**10 Example**

See the example programs associated with the problem solving function you wish to use for a demonstration of how to use `nag_fit_opt_get` (e02zlc) to query options.

---